In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institute shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the Dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

7/25/68

A MODEL OF THE ECONOMICS OF LARGE, DISCRETE-EVENT DIGITAL COMPUTER SIMULATION

A THESIS

Presented to

The Faculty of the Division of Graduate

Studies and Research

By

Ronald Edwin Rezek

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in the School of Industrial and Systems Engineering

Georgia Institute of Technology

December, 1972

A MODEL OF THE ECONOMICS OF LARGE,
DISCRETE-EVENT DIGITAL COMPUTER SIMULATION

Approved:

Donovan Young

Gerald J. Thuesen

Paul Gray

Date approved by Chairman: 9/14/72

ACKNOWLEDGMENTS

This author wishes to express his sincere gratitude and appreciation to Dr. Donovan Young for his willingness to assume the committee chairmanship in the middle of the research and for his guidance and invaluable assistance which enabled this work to be completed. I thank Dr. Paul Gray for providing the initial impetus for this research and for remaining on the committee to assist in the completion of the research. My thanks to Dr. Gerald Thuesen for his research ideas and writing assistance.

Last, but certainly not least, I thank my wife, Linda, for her assistance in preparing the various drafts. Her patience, understanding, and constant encouragement were instrumental in enabling this work to be completed. To her, I affectionately dedicate this work.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Continued)

SUMMARY

This thesis presents a decision-flow method for gathering and analyzing the relevant information needed to make development decisions regarding large-scale discrete-event simulation studies--such decisions as what method of data collection to use, whether or not to use a preliminary model, what simulation language to use, and whether to use outside consulting services at various stages.

The approach is that of minimizing the time-discounted expected cost of all aspects of the simulation study--not only programming time and computer time, but also data collection, preliminary model development, model validation, and analysis and implementation of the simulation results.

The decision-flow model, which was designed to include the decisions identified as important in the available literature, was validated experimentally by applying it to actual large-scale simulations. Narrative histories were collected from each of two organizations at a time when data on the decisions made, the reasons for the decisions, the costs of the chosen alternatives and the estimated costs of the rejected alternatives were still freshly available. Retrospective applications of the decision-flow model to the actual decisions made in these two simulation projects are presented in detail.

CHAPTER I

INTRODUCTION

Managers, particularly those at the highest levels, study or analyze complex situations and problems. Because solutions to many such problems can not always be determined analytically, simulation has been finding acceptance as an aid to decision making. Discrete-event simulation was developed to meet the need for a quantitative method of analyzing some of these complex situations.

"Simulation" herein will refer to discrete-event simulation, which from its earliest form (the Monte Carlo methods used by Ulam, Fermi, and Von Neuman) (5) has been closely associated with the development of the digital computer. The speed with which the computer can perform mathematical operations and its rapid access memory make it a practical tool for use in simulation applications. The result has been a rapidly advancing state of the art of simulation.

Simulation is one of several methods of comparing alternatives available to the decision maker. The decision maker must answer two distinct but interrelated questions with regard to simulation for each problem: (1) should simulation be used as the solution method; and (2) if so, how to simulate. The method of simulation chosen can affect the answer to the first question.

The decision to simulate is not a simple yes or no decision. An analysis of alternate solution methods should be conducted (14). During this analysis, the decision maker should try to learn as much as possible about the problem. He should compare the resource, time, and money requirements to the worth of the benefits of each alternative. Use of this analysis will assist him in deciding for or against the use of simulation.

Having made an affirmative decision to simulate, the decision maker concentrates on the simulation, its development, and ultimate use. This requires him to make decisions which affect the development of the resultant simulation.

System Overview

Simulation is often thought of by the manager as the program run on the computer. As a result, he focuses most, if not all, of his attention and effort on the costs and resource requirements associated with the computer and the computer program. The manager has used the computer primarily for rapid access to information and is acutely aware of the cost of computer run time. However, the computer run is only a part of the simulation process.

In general terms, the simulation process has four major stages. As shown in Figure 1, these are: (1) Development, (2) Operation, (3) Modification, (4) and Repeated Operation. Stages 3 and 4, Modification and Repeated Operation, may be iterative, or may never be reached, depending on the objectives sought. The costs associated with each stage result from the decisions made within the stage.

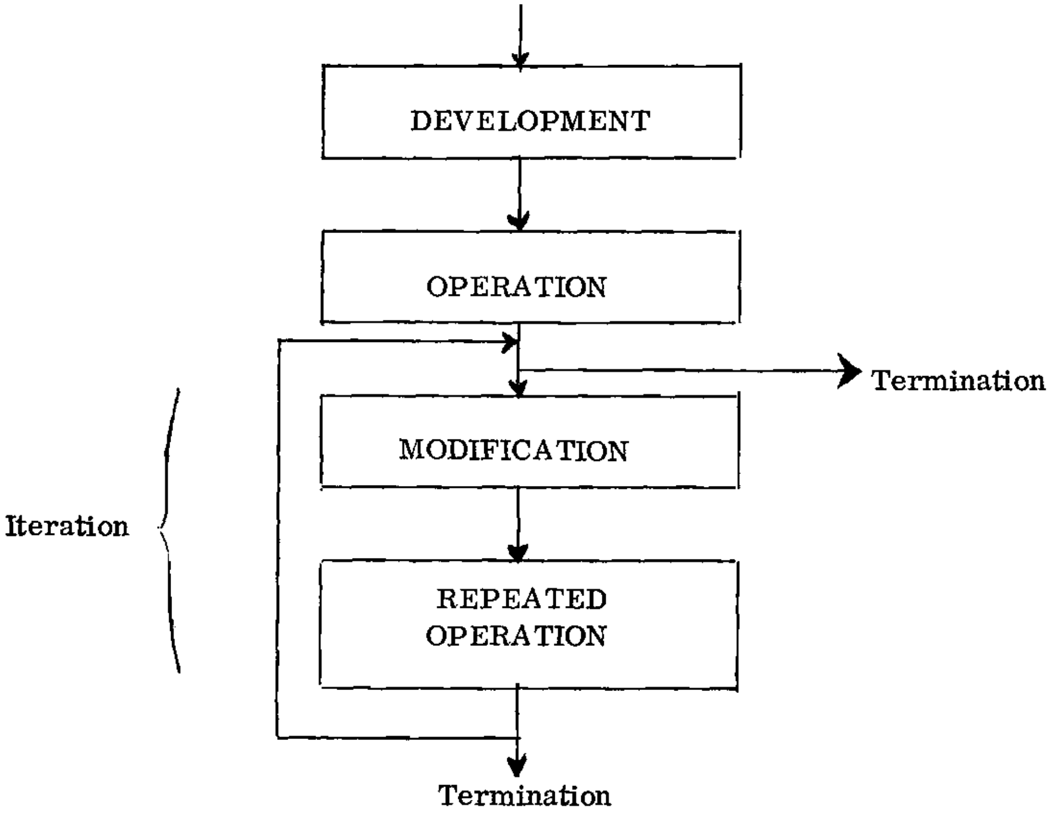

Figure 1. Four Stages of Simulation

This thesis concentrates on modeling the development stage, but includes structuring the study of the economics associated with all stages. The development stage, or process, consists of the activities shown in Figure 2. These activities are: Problem Definition; Data Collection and Processing; Mathematical Model Development; Computer Program Development; Design of Experiments; Computer Operation; and Analysis and Use of Simulation Results. Thus, the decision maker who focuses his attention on the computer and the computer program is concentrating on only a part of the simulation development process. In each component, decisions are made which affect or shape the resultant simulation. In each decision, the economics associated with the alternate courses

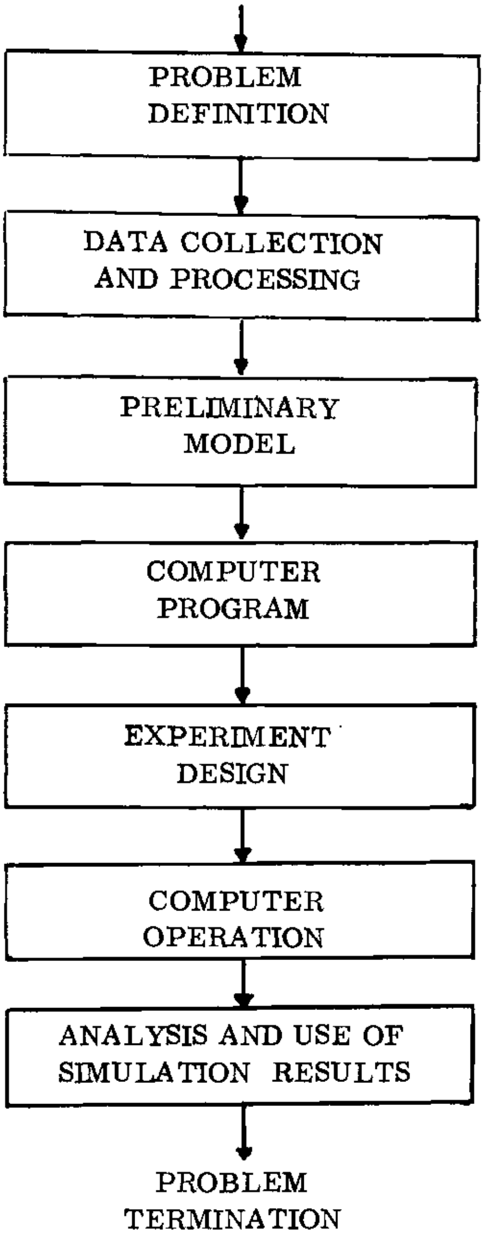

Figure 2. Sequence of Activities in a Simulation Development.

of action must be considered.

Consider, for example, the issues in selecting the simulation language. One language (FORTRAN)* may require more programming time (costs), but less computer time (costs) than another (GPSS)*. The other language (GPSS), while requiring less programming time, would necessitate training (costs) for the computer programmers. These cost trade-offs must be considered when making decisions in the simulation development process.

Procedure and Objectives

The primary objective of this thesis is to develop a model of the digital computer simulation development process that will enable analysis of the associated economics.

Secondary objectives include the application of a cost methodology to the model developed and validation of this model by case examples.

This thesis is a first step in the study of the economics of simulation. It provides a framework to the simulation user for analysis of the economics associated with decisions in the development process.

Scope

The scope of this thesis is limited to large, discrete-event digital computer simulations. "Discrete systems include variables that can take only particular values from among a finite (but possibly very large) set of

---

*FORTRAN, GPSS and all other languages mentioned herein are described in elementary simulation textbooks such as Refs. 6, 14, and 20.

alternatives" (14). The definition of 'large', for this thesis, is arbitrarily set to include those simulations which have a total development time greater than two man-years or have a run time equivalent to fifteen minutes on an IBM 360/50 series computer.

CHAPTER II

LITERATURE SURVEY

The techniques and philosophy of simulation have been extensively discussed in a wide-ranging literature. Specific topics range from those which relate to one aspect of simulation to those which describe the entire development process. Within the area of discrete-event digital computer simulation, the literature can be separated into two specific areas: the development of a simulation and the economics of simulation development. This separation is necessary, despite some overlap and interaction between the two areas, in order to define and structure the process of simulation independent of the economic considerations associated with the development process.

Development of a Simulation

Definition of Simulation

Simulation has its origin in the study of systems. A system can be defined as "...a collection of regularly interacting or interdependent components...acting as a unit in carrying out an implicitly or explicitly defined mission" (14). Discrete systems, those considered in this research, can be adequately described as a series or chain of events. Typical systems are: a customer in a grocery store, a transportation system, and a telephone switchboard.

Analysis of a system usually requires that a model, a representation of the system, be developed and used. Certain features of the system may be eliminated, abbreviated, changed, or approximated in the model. Two models of the same system need not be identical. However, each can satisfy the needs of the user.

Simulation has been defined in a variety of ways. Mize and Cox define simulation as "...the process of conducting experiments on a model of a system..." (5). Ockene states his definition as "...the construction of a dynamic model of a system and its operation on a computer (19). Naylor describes simulation in somewhat more detail as "...a numerical technique for conducting experiments on a digital computer which includes...models that describe the behavior of a system over extended periods of real time" (20). Kiviat, Tocher, Gordon, and Schmidt and Taylor present similar definitions (17, 24, 15, 23). In each definition, the key ideas of system, model, time, components, model operation, and experimentation are present.

This research will use the definition presented by Hillier and Leiberman who state that simulation is "...the technique of performing sampling experiments on a model of the system" (16). The simulation 'run' consists of a number of simulation experiments. Each experiment is a realization of a complete chain of events.

Development Process of a Simulation

The complete simulation development process is presented through case studies or examples, hypothetical or real, by several authors (6, 7, 22). These examples served as additional informal validation data (beyond that provided by the two application studies reported in Chapter V) for the decision-flow model developed in Chapter III and Chapter IV.

The major steps in the simulation development process are presented by Appelbaum, Naylor, Maisel and Gnugnoli, and Emshoff and Sisson (2, 20, 14, 6). A modification of these will be used in the model development in Chapter III.

Simulation Languages

Emshoff and Sisson state that there are two levels of decision making in selecting a simulation language (6). Level one concerns selection of the languages to be made available on the computing machinery. Level two concerns the problem analyst who must choose one of these languages for use in a particular simulation. The model developed in Chapter III is directly applicable only to those decisions made at level two. However, the model may be extended by assigning a negative cost to the future benefits of having a new language added to the system.

Chubb presents an economic evaluation of a particular simulation language (4). His procedure could be applied to other languages, and is applicable to the simulation development process. However, it omits several cost factors considered here, namely the costs of data collection, preliminary model development, model validation, and implementation.

Teichroew and Lubian (18) presented a comparison of several computer languages in 1966. They did not offer a formalized method of making decisions about developing a simulation model, but they did offer a cogent discussion of the important trade-off between programming cost and execution cost. They found specialized languages such as GPSS to be low in programming cost but high in execution cost, general languages such as FORTRAN to be high in programming cost but low in execution cost, and partly specialized languages such as SIMSCRIPT (actually a general language with specific powerful simulation features) and GASP (a set of simulation-specific FORTRAN subroutines) to be intermediate in the two costs.

Justification and Rationale

Simulation is a method for comparing alternatives. It may be used, as Hillier and Leiberman state, when problems are so complex that analytic solution is impossible and simulation provides the only practical solution approach (16). "However, systems do not have to be large or complex for simulation to be useful," as Kiviat states (17). Regardless of problem size and complexity, other reasons justifying use of simulation are: (1) a requirement for experimental control; (2) a need to maintain a constant environment during experimentation; (3) a need to study the real world system without modifying it in any way; and (4) the fact that a simulation model, once developed, can be used as often as desired (14, 17, 23).

Disadvantages of Simulation

Some disadvantages of computer simulation are: (1) it generally takes a long* time to develop; (2) it requires scarce and expensive resources, such as, computer time; (3) it may require extensive field studies; and (4) it may hide critical assumptions (14, 23).

---

*For simulations of the size treated herein, two man-years is a typical development time.

An additional disadvantage often results from a combination of human nature, the desire for excellence, and the pressures of competition. These human factors combine and cause simulation developers to make simulation models as detailed as possible. Besides increasing development time and cost, the model often produces unnecessary or unmanageable data.

The advantages and disadvantages should be considered when making the decision to use simulation as the solution method. The assumption in this thesis that simulation is the solution method selected implies that these advantages and disadvantages were considered either explicitly or implicitly by the decision maker before using the results of this thesis.

Economics of a Simulation

Literature concerning the economics of digital computer simulation is rather scarce. Simulation is often described as expensive (14).

A general survey, covering approximately fifty large discrete-event digital computer simulations, was conducted by Abt Associates, Inc. (1). The simulations were divided into groups based on the total cost and the total time of simulation development. Total cost and total time are considered, but no further breakdown is provided.

Fried presents a method of computer project cost analysis (11). The method, a cost, benefit, and cash flow analysis, is similar to those used for capital investments. Operating and implementing costs are divided into their component costs. Three estimates, pessimistic, most likely, and optimistic, are made for each component. Payback and cash flow analysis result from manipulation of these cost estimates. Alternative solution methods to a particular problem are compared by this analysis method.

The method required one man-year of systems analyst time to complete analysis of the example presented, which had a project cost of between \$500,000 and \$1,000,000. A guideline amount of \$25,000 was presented as the minimum project cost to warrant this type of analysis. Fried's work differs from this work chiefly in that Fried's method is not specifically developed for simulation cost analysis.

Economic analysis models generally involve decision analysis as presented by Raiffa (21). The primary concern is the economic impact of decisions. One of the best known methods for dealing with decision problems is the decision tree. This method, presented by Raiffa, is an easily followed sequence of steps which requires only a fundamental knowledge of probability. The decision tree approach is used in this thesis.

CHAPTER III

MODEL DEVELOPMENT

Model Components

The simulation development process is, in effect, a system. The system parts interact to allow accomplishment of a set of requirements, the simulation objectives. The major components or steps of a computer simulation development have been identified by Appelbaum, Naylor, Maisel and Gnugnoli, and Emshoff and Sisson (2, 20, 14, 6). These are shown in Figure 3. Although the components identified by each author differ in number and specific title identification, they are similar in content and follow in approximately the same general chronological order. The components presented by Naylor contained most of the components necessary for this research. However, this list was modified somewhat to provide the breakdown needed to study the costs associated with the development process. The components selected are shown in Figure 4. This flow chart describes the sequence of major components and the feedback loops that result from additional information.

Events and End Items

Each of the major components is divided into several large parts called <u>events</u>, which are accomplished in a specified sequence. Events often include participation by personnel from more than one department within the firm or

Appelbaum (2)	Naylor (20)	Maisel and Gnugnoli (14)	Emshoff and Sisson (6)
1. Problem Definition	1. Formulation of the Problem	1. Preliminary Analysis	1. Define the Problem
2. Simulation Decision	2. Collection and Processing of Real World Data	2. Formulation of the Problem	2. Analyze Data Requirements and Data Sources
3. Date Collection and Reduction	3. Formulation of Mathematical Model	3. Collection and Analysis of Pertinent Information	3. Formulate Models of Sub-systems
4. Model Development	4. Estimate of Parameters	4. Model Construction	4. Combine Sub-systems into Simulation Model
5. Model Evaluation	5. Evaluation of Model and Parameter Estimates	5. Computer Programming	5. Gather Data and Estimate Parameters
6. Simulation Programming	6. Formulation of a Computer Program	6. Validation	6. Program and Debug Simulation
7. Analyzing the Simulation	7. Validation		7. Validate Simulation Model
	8. Design of Simulation Experiments		8. Design Simulation Experiments
	9. Analysis of Simulation Data		9. Run Simulation
			10. Analyze Results and Present to Management
			11. Implement Results

Figure 3. Major Components of the Development Process of Digital Computer Simulation.

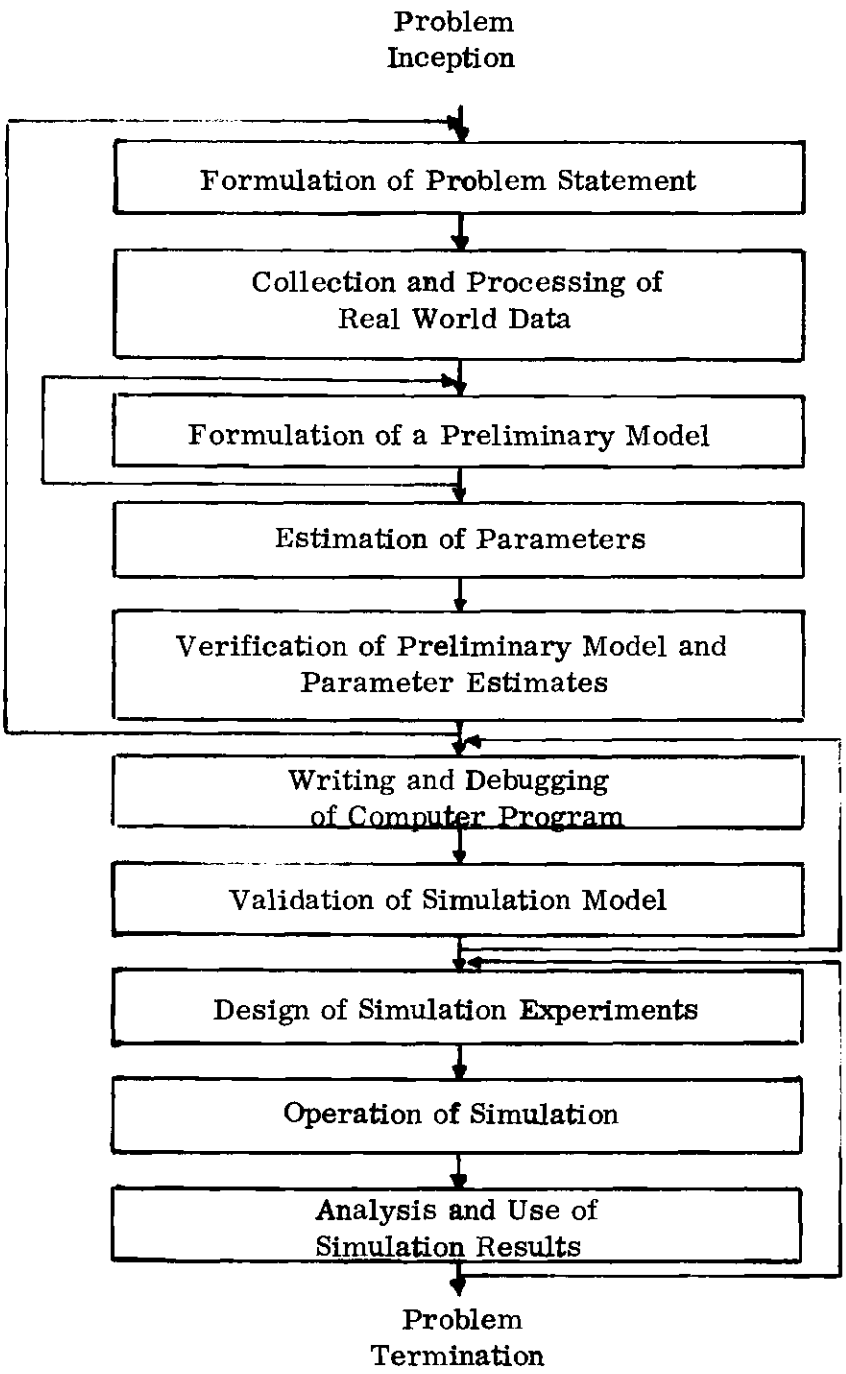

Figure 4. Flow Chart of Major Components of Computer Simulation Development Process.

organization as well as resources from more than one source. For the decision maker, this can make selection of a particular alternative course of action within an individual event difficult. Therefore, the events have been divided into smaller parts, called end items. End items are basic work packages of the development process that are limited in scope to jobs which are directly responsive to one middle manager or supervisor.

Example

The structuring process is shown in graphical form in Figure 5. Simulation consists of four stages. The development stage consists of ten major components. The first component, Problem Formulation consists of two events; problem specification and objectives definition. The event, Objectives Defined, consists of three end items; specify objectives identify outputs, and specify relevant variables.

Decision-Maker Explanation

Development of a computer simulation requires a variety of skilled individuals; operations research analysts, statisticians, system analysts, and data processing specialists. These individuals may be the decision makers responsible for choosing a particular course of action during the simulation development process. The number of decision makers and their exact areas of responsibility is determined by the individual simulation being developed and the organizational structure of the developing firm. In some cases, one individual will be the decision maker throughout the development process. In other cases, the term

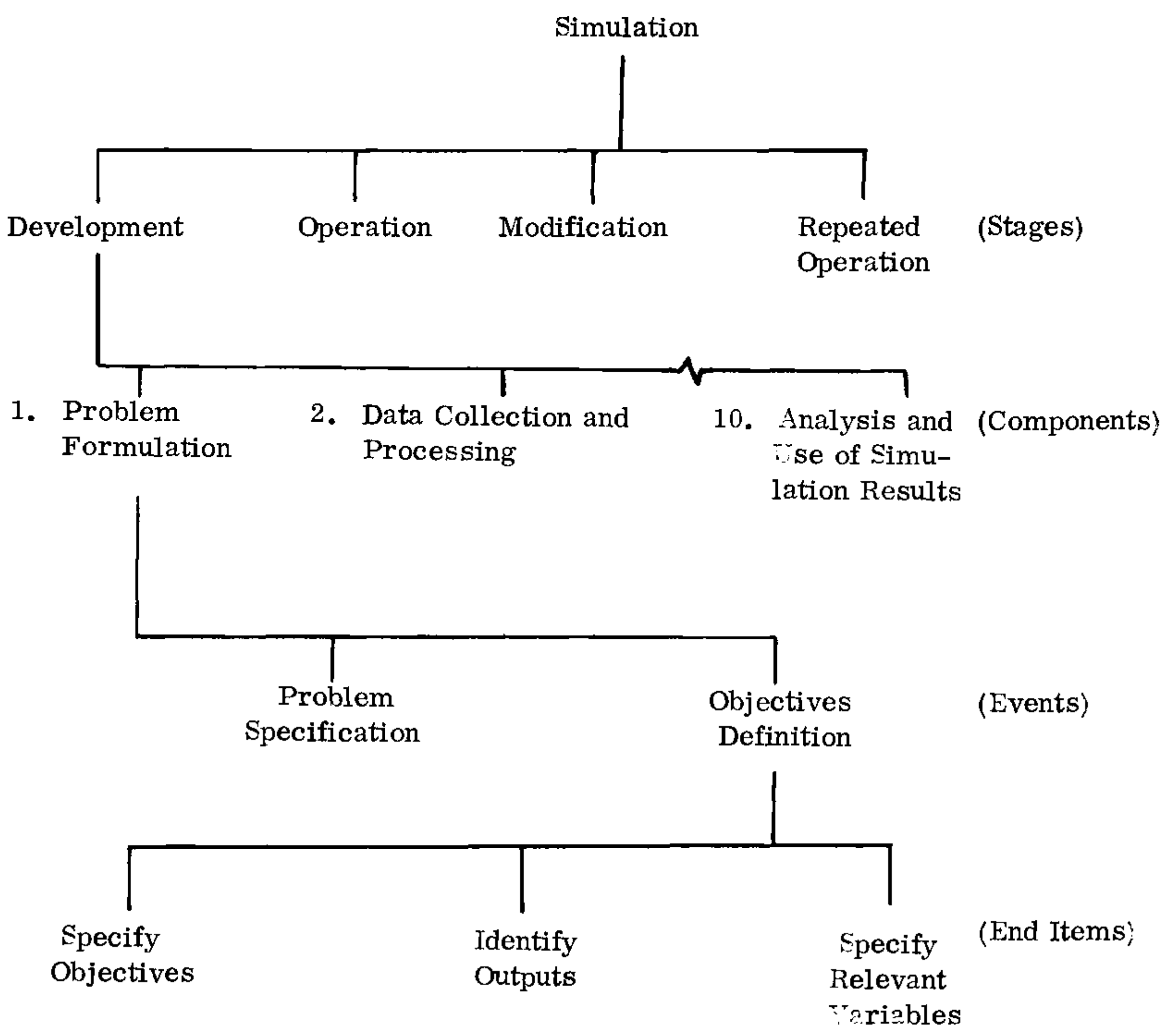

Figure 5. Structuring Process of Developed Model.

decision maker will refer to many individuals. The simulation developed by the decision makers and their staffs is for use by a high-level manager as an aid in analyzing a complex situation.

The next section presents and explains the events and end items of each major component.

Explanation of Components

The general explanation of the procedure used to develop the model presented the major components of the simulation development process and the method used to designate events and end items.

This section presents in sequential order the events and end items of each major component. Emphasis is placed on the alternatives facing the decision maker. The model developed is exhaustive. Not all components, events, and end items presented occur in all simulations.

The diagram format used to present the components in this section is shown and explained in Figure 6. Figure 6 is a key for reading Figure 7 through Figure 16; in these latter figures, each end item is specifically named.

Detailed Formulation of the Problem Statement--Component 1

Once the initial decision has been made to use simulation as the solution method for a given problem, the problem is usually reviewed and refined. This refinement is a continuing process because as more and more become known about the problem, it is possible to sharpen the definition. The refinement process can lead the decision maker to one of three alternatives: (1) He continues to accept the problem as valid and simulation as the solution method; (2) he initiates

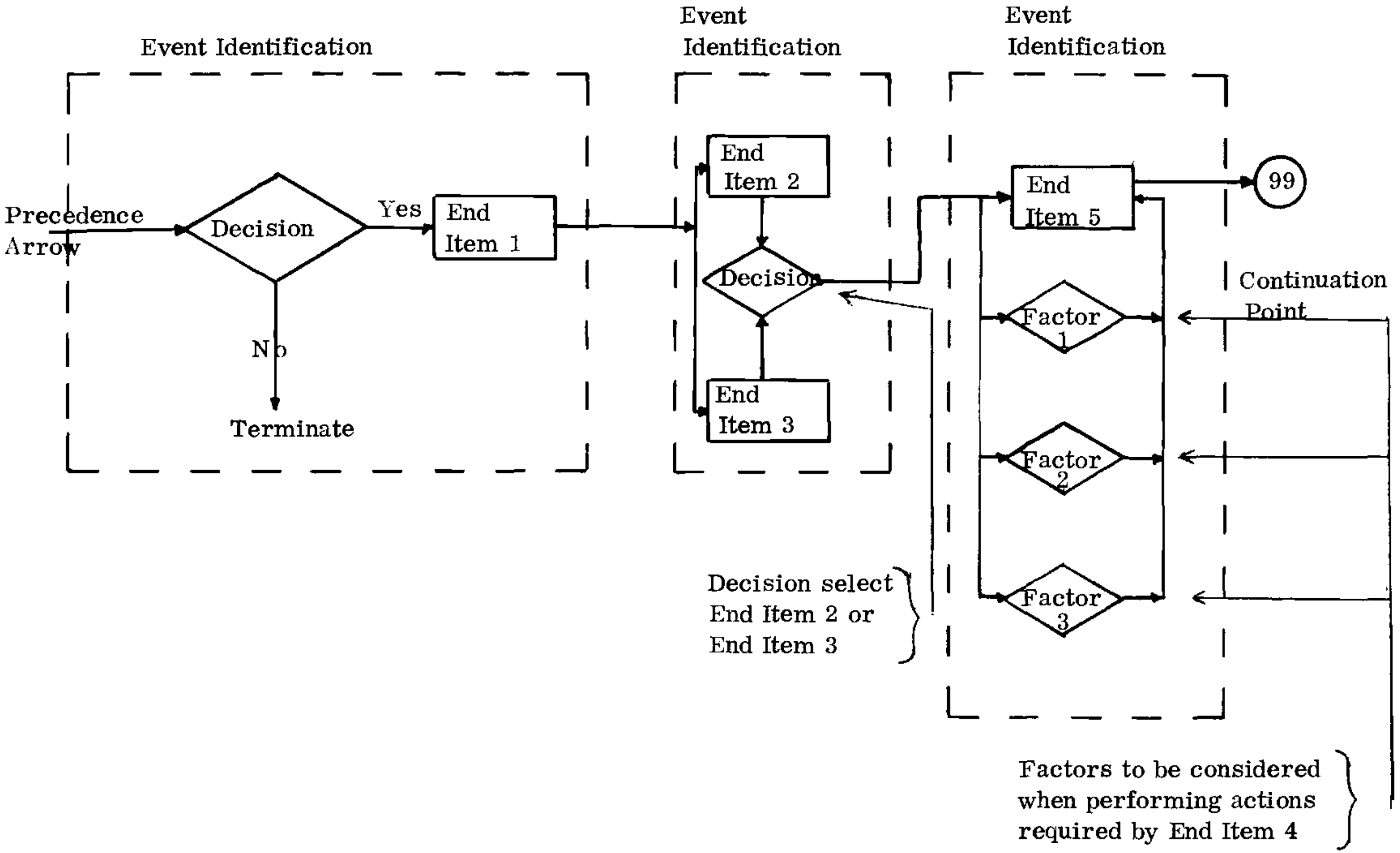

Figure 6. Diagram Format Explanation.

additional study which may result in a revised problem statement; or (3) he finds that the problem can best be solved by a method other than simulation or should be discontinued. This is shown in Figure 7. If the simulation is not aborted, the next end item, Problem Refinement, follows automatically. The problem is likely to be vaguely defined in qualitative terms, which must be translated into operational terms. This refinement gives management increased insight into the problem and is essential for the beginning of the modeling effort.

Definition of the objectives of the simulation includes three end items: (1) specification of the objectives, (2) specification of the relevant variables, and (3) identification of the outputs desired. These three end items, while separate, interact to define the objectives of the simulation. The relevant variables are those which affect system performance as measured by the specified objectives. This performance measurement is reported in the outputs selected.

The costs associated with this component may not be large compared to those of other components. However, formulation of the problem is the most crucial step in developing the simulation model of the system (6). Successful accomplishment of the simulation objectives depends on correct problem formulation.

Collection and Processing of Real World Data--Component 2

The collection and processing of real world data component is shown in Figure 8.

The problem statement requires data (quantitative or otherwise) for use in developing the mathematical model, simulation experiments, and model

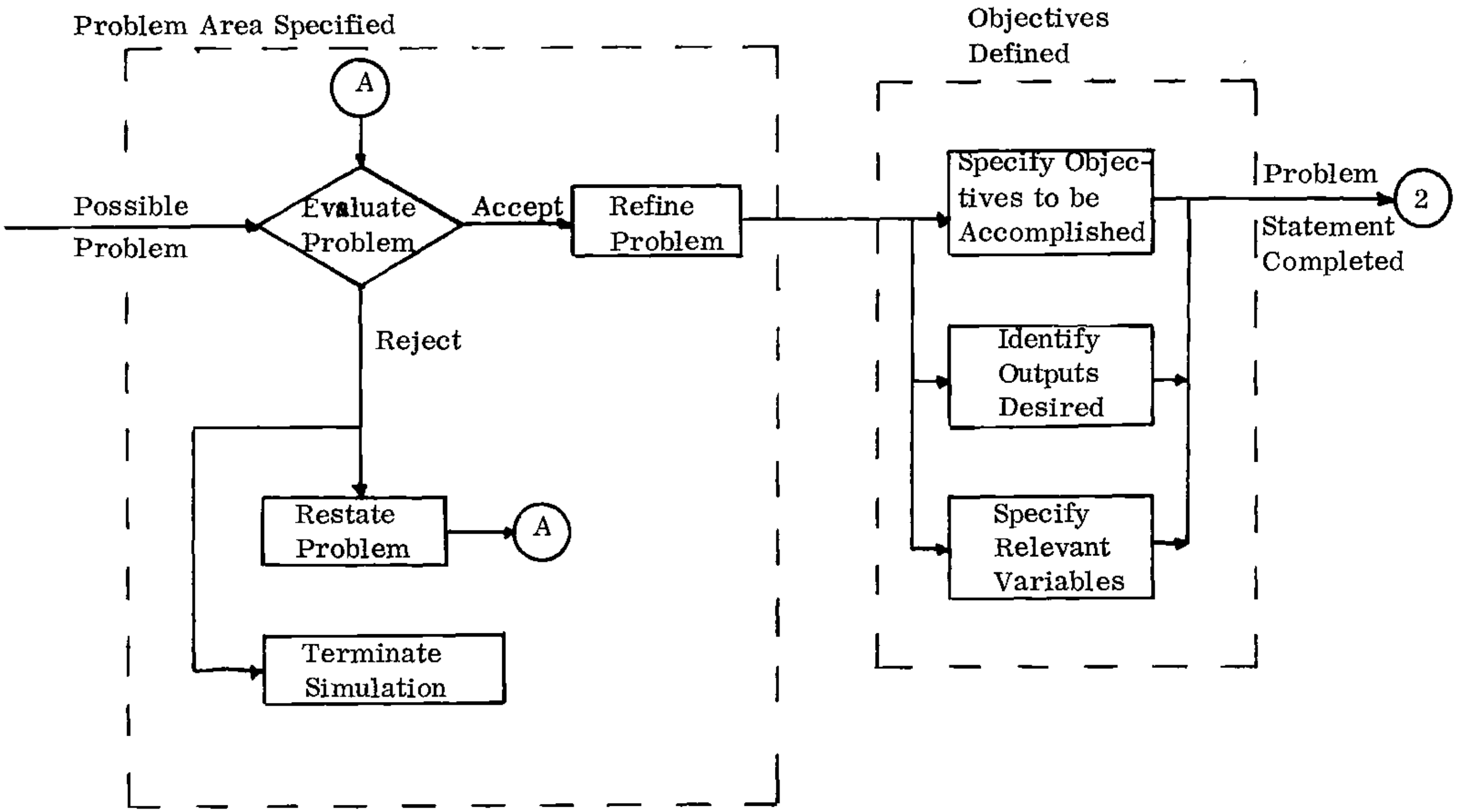

Figure 7. Formulation of the Problem Statement--Component 1.

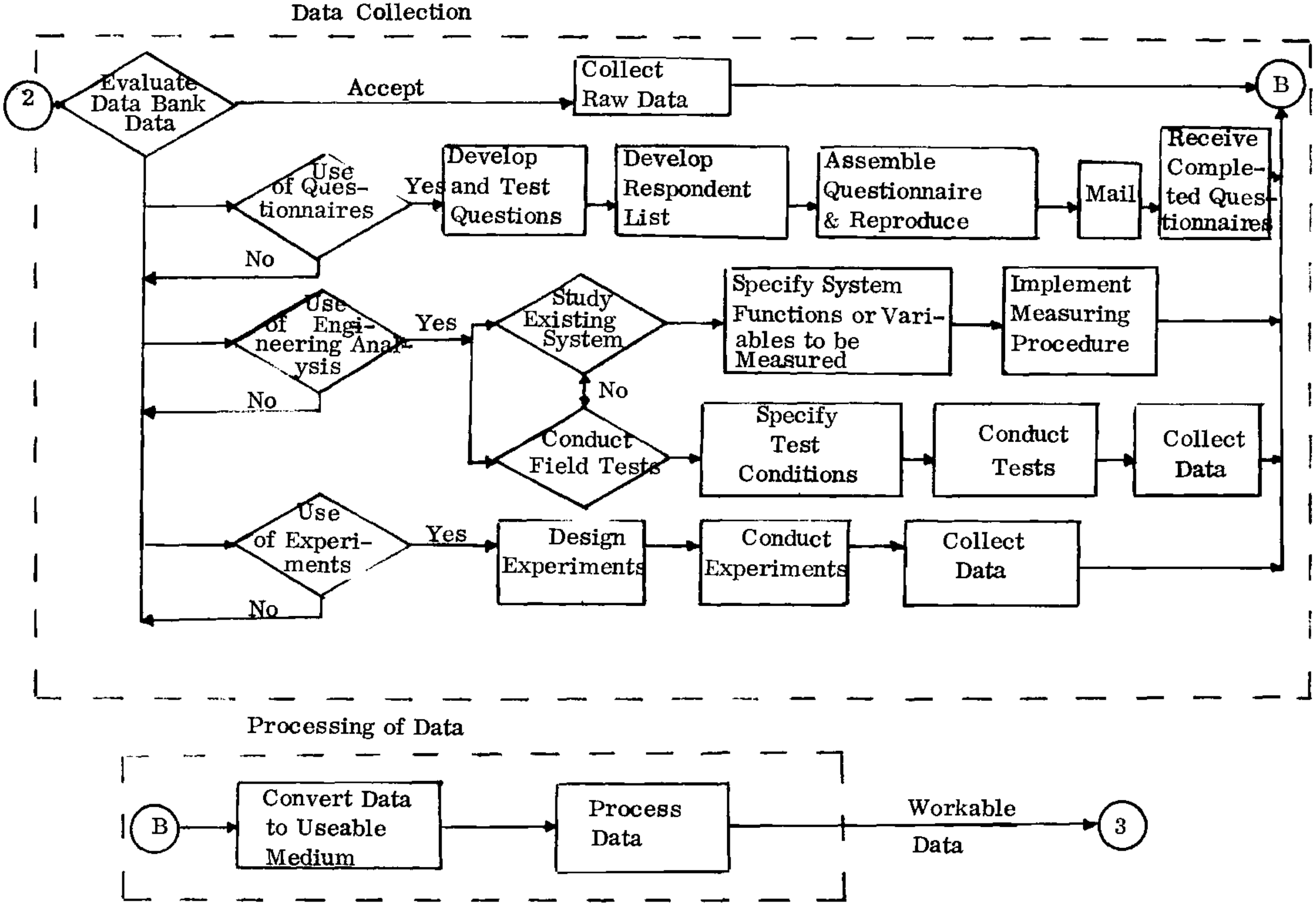

Figure 8. Collection and Processing of Real World Data--Component 2.

validation. The data collection activity begins with the review and evaluation of the available data. If information in the data bank is available, processing of the data can begin. If it is not available, a data gathering phase must be undertaken. The methods of data gathering include use of questionnaires, interviews, engineering analysis, experiments, or tests. The data required is specified by the decision maker. The applicable end items follow automatically with the selection of a data gathering method.

For example, if questionnaires are to be used, an appropriate set of questions must be developed and evaluated. The questionnaire is then assembled, reproduced, and sent to a list of selected respondents. The completed questionnaires returned by the respondents is now available as raw data for processing.

Processing of the data begins when raw data is collected. The data must often be converted from one medium to another. For example, handwritten documents must be punched on cards, which may be converted to magnetic tape, if the data is to be used by a computer. Conversion of the data allows data manipulation, which produces the workable data.

The workable data achieved from this component is used in the development of the mathematical model, the next component, and in simulation experiments and model validation.

Development of the Preliminary Model--Component 3

The development of the preliminary model component is shown in Figure 9.

Model formulation begins with identification of the subsystems which make up the system to be modeled. These subsystems reduce the large, complex system

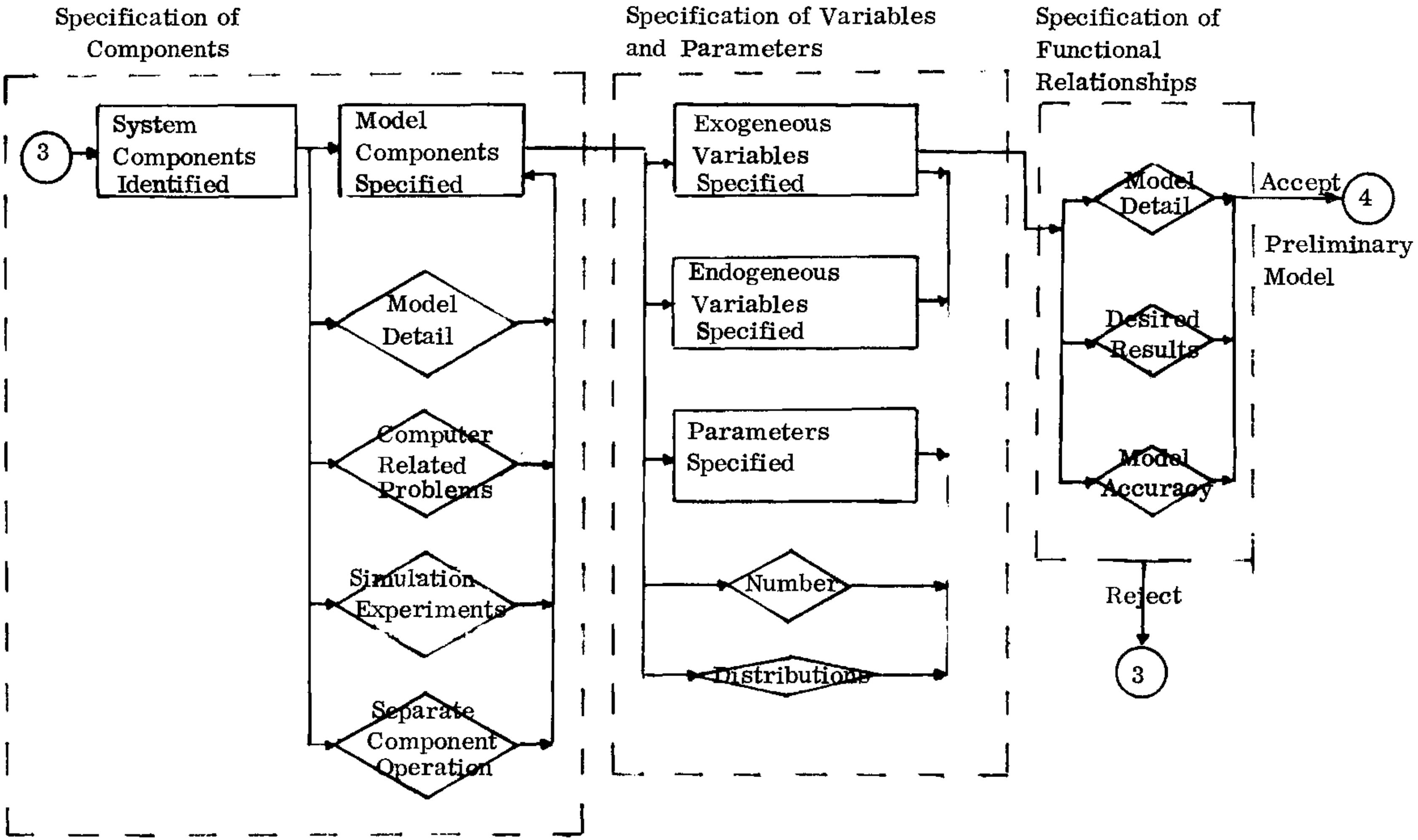

Figure 9. Development of the Preliminary Model--Component 3.

into less complex components, which makes modeling less difficult.

Factors considered during the component specification process are: (1) the amount of detail desired in the model; (2) programming time, program debugging time, and computer efficiency; and (3) the simulation experiments to be conducted.

Sometimes, useful information can be gained by operating a simulation model of a subsystem, with no necessity for operating the entire simulation model. This alternative should be considered, but generally is not.

Specification of the model components allows the specification of the variables, exogenous and endogenous, and parameters to begin. The number and choice of the input variables is often difficult. Too few input variables may cause less model detail to be achieved or may cause the model to be invalid. Too many input variables can reduce computational efficiency and increase model complexity. The output variables generally specified in the simulation's objectives, are dependent on input variable specification. These output variables must be achievable through use of the specified input variables. Parameter specification is closely tied to variable specification. The parameters must enable the input variables to produce the specified output variables.

The functional relationships which unite the variables and parameters within the specified components, produce the preliminary model and enable it to provide the desired results. The functional relationships should not increase model detail, cause production of unnecessary results, or produce an inaccurate model. Acceptance of the specified functional relationships, as constrained by the modeled system's characteristics, results in achievement of the preliminary model. Rejection of the functional relationships requires repeating this component until acceptable results are achieved.

Estimation of Parameter Values--Component 4

The estimation of parameter values component is shown in Figure 10.

Estimation of the parameter values consists of selecting an estimation method, verifying that the selected method can produce the desired results, and applying it to the historical data. The estimation methods, which include use of ordinary least squares, single equation techniques, simultaneous equations, and collection of values over time to compute a frequency distribution, produce the parameter value estimates.

The parameter values are then tested. If the results of the tests are satisfactory, the estimated values are accepted. If the results are rejected, estimation and testing is repeated until acceptable parameter values are achieved.

Verification of the Preliminary Model and Parameter Estimates--Component 5

The verification of the preliminary model and parameter estimates component is shown in Figure 11.

Before testing can begin, model preparation, which includes selection of the tests to be conducted, specification of the accuracy levels, and identification of the portions of the model to be tested, is accomplished. Test selection includes consideration of the results achieved from each test, time required to conduct the tests and analyze the results, and test applicability. The tests include tests of means and variances, tests based on count data, and nonparametric tests.

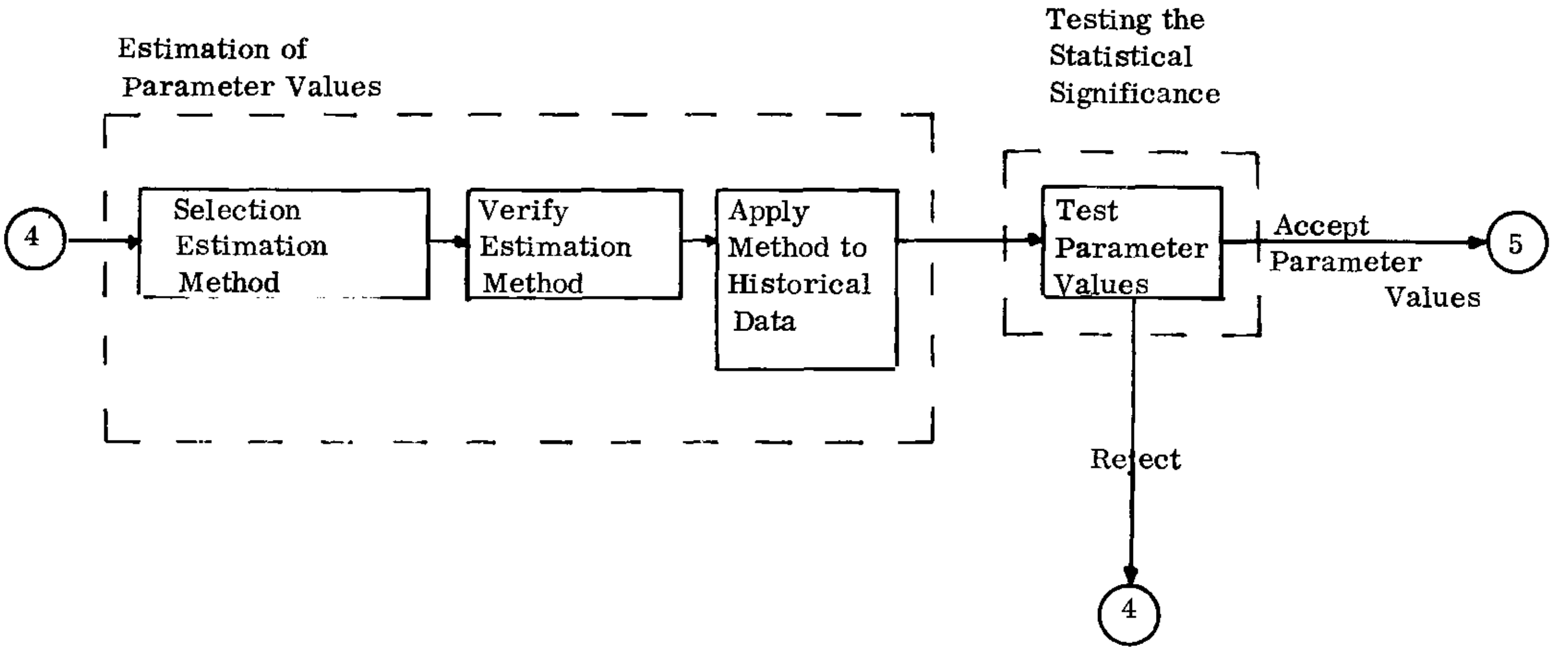

Figure 10. Estimates of Parameter Values--Component 4.

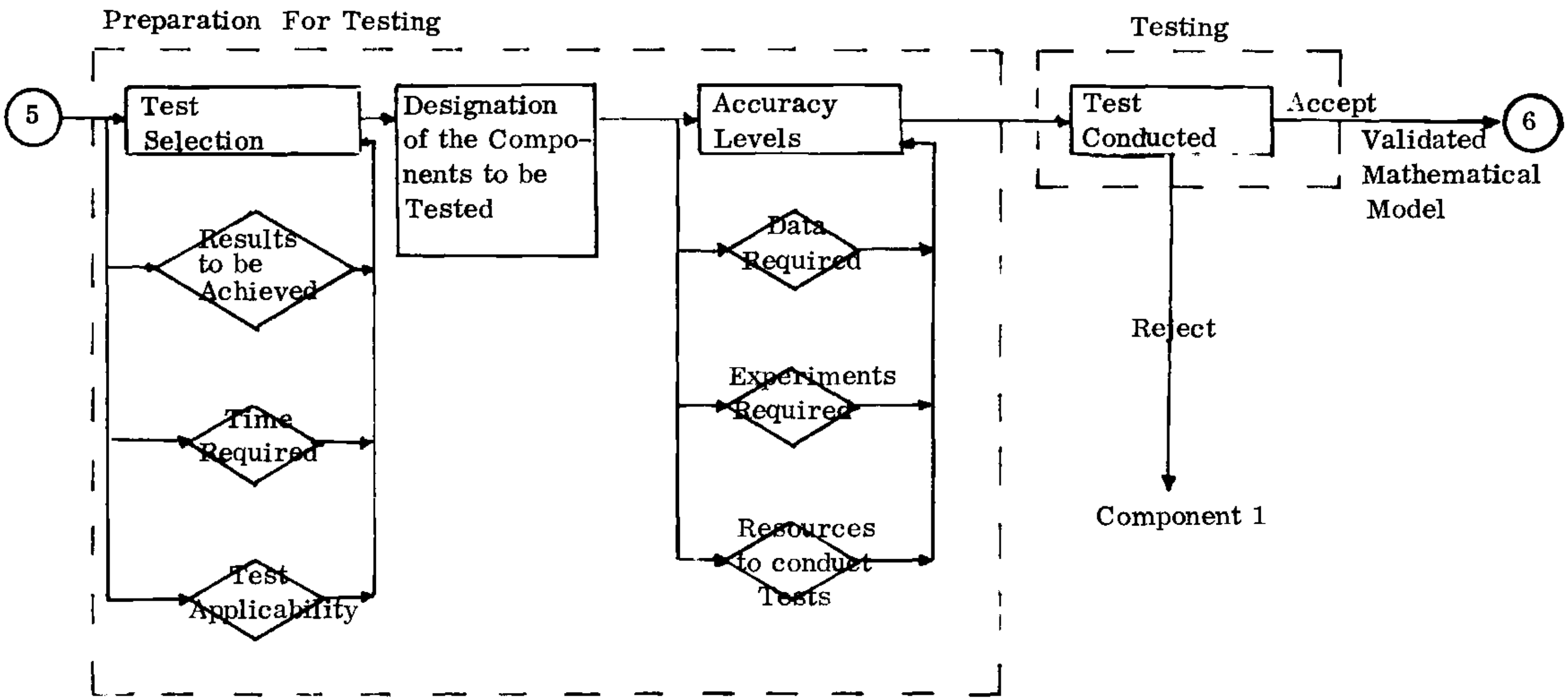

Figure 11. Verification of the Preliminary Model and Parameter Estimates--Component 5.

Designation of the model components to be tested is done in conjunction with test selection. Evaluation of all components may not be necessary if validation can be achieved by testing selected components. Reducing the size of the model to be tested can reduce the costs.

Specification of the accuracy levels is the third end item in the preparation event. Consideration must be given to data requirements, number of experiments, time, personnel, and equipment necessary to conduct each test. Benefits achieved by increasing accuracy levels must be compared with the additional costs required to obtain these increased levels.

Completion of the testing preparation allows the designated tests to be conducted. Accuracy of the conduct of the tests versus the costs, personnel, time, and equipment, should be considered. Completion of the tests is followed by analysis of the results. Acceptance of the results is acceptance of the model as a valid representation of the system. Rejection of the results signifies that Components 1 through 5 should be repeated. This may seem startling because of the costs involved in repeating these components. However, an invalid model which can not produce valid results is of no value to the simulation developer.

Writing and Debugging of Computer Program--Component 6

The writing and debugging of the computer program component is shown in Figure 12.

Formulation of a flow chart, which describes the steps of the computational procedure and the relationship between them, requires the decision maker to specify the amount of flow chart detail necessary for his particular simulation.

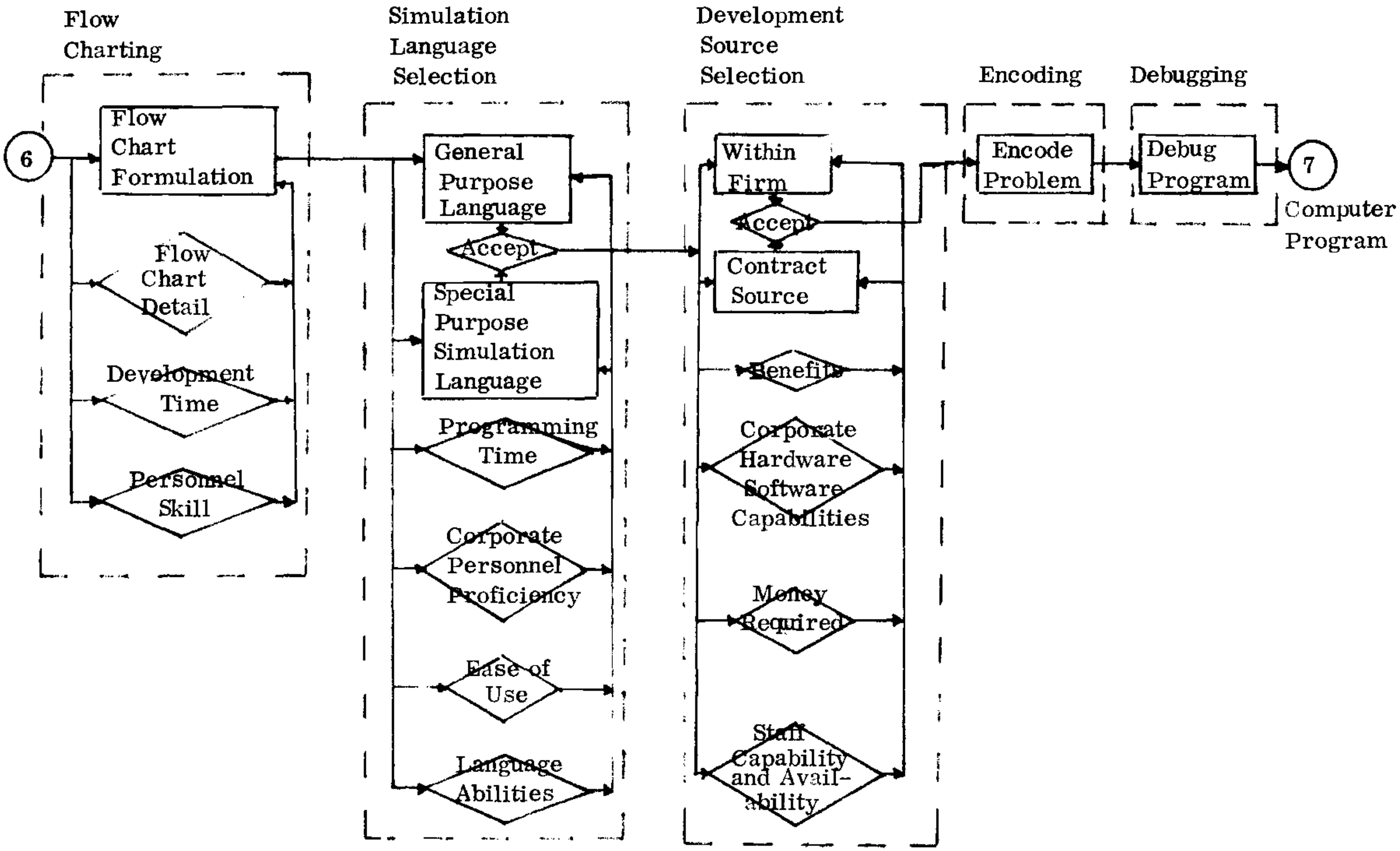

Figure 12. Writing and Debugging of Computer Program--Component 6.

Increased detail requires more development time and more highly trained personnel. However, this increased detail can reduce time, personnel, and personnel skill necessary for problem encoding. The decision maker uses these trade-offs when specifying flow chart detail.

The decision maker is faced with two general considerations when selecting a simulation language. These are: (1) the question of a general purpose language versus a special purpose simulation language; and (2) contract program development versus development within the firm. The decision concerning the source of program development includes consideration of: (1) corporate computer hardware and software capabilities; (2) staff capabilities and availability for the encoding task; (3) time and money requirements; and (4) specific benefits achieved from each source.

The decision concerning general purpose or special purpose simulation language requires consideration of: (1) programming time, which is generally shorter for simulation languages; (2) computer running time which is generally longer for simulation languages; (3) the training required for corporate personnel to be proficient in a specific language; (4) the ability of the language to provide the desired results.

Large organizations often place an over-riding constraint on this decision-making process. This constraint limits the program development process to corporate personnel who must use available computer equipment and languages. As a result, an important decision-making process is often ignored.

Problem encoding requires a general encoding plan. Some examples are:

(1) use all programmers to encode the problem from beginning to end; (2) select individual components to be encoded; and (3) split the programmers and encode the problem at several points simultaneously. Numerous variations are possible. Encoding, debugging, and validating one or more components of the model separately sometimes enables results to be achieved from operation of this portion prior to encoding the remaining components.

Debugging the program, which is often aided by error detection diagnostics included in the computer language, is complete when the program is identical to the original flow chart. The debugged program is ready for validation.

Validation of the Computer Simulation Model--Component 7

The validation of the computer simulation model component is shown in Figure 13.

Validation of a computer simulation is a difficult problem (20). As such, validation is not always accomplished. Validation may be: (1) Comparing the simulated data with historical data; (2) Checking the performance of model subsystems; (3) Checking the variance of the outputs when the inputs are held constant (high internal variability can obscure changes in outputs which resulted from changes in inputs); or (4) Asking people who know the real system to judge whether the model is reasonable (this is not scientific validation, but a test of reasonableness) (20, 6, 24).

Validation begins with preparation for testing. Test selection and model preparation procedures, presented in detail in Component 5, are not repeated here. Test accuracy levels are determined by intended use of the simulation

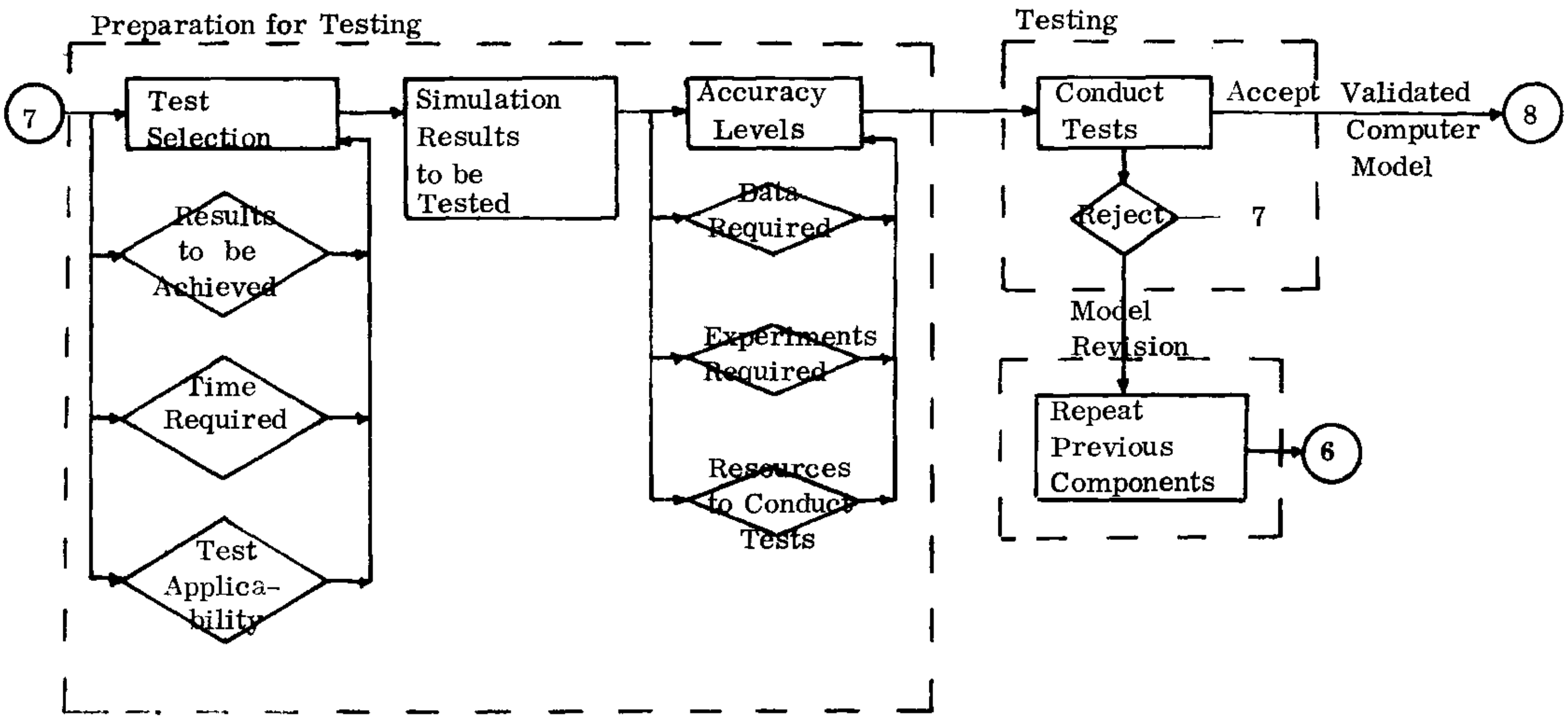

Figure 13. Validation of the Computer Simulation Model--Component 7.

results. Lower accuracy levels are advantageous when the benefits achieved from higher accuracy levels can not be implemented in the real world system.

Validation testing is conducted on the computer simulation model as it will be operated in general use. The decision maker accepts or rejects the test results. Acceptance means the model is ready for use. Rejection means that the tests need to be repeated or that the model requires revision.

Design of Simulation Experiments--Component 8

Design of computer simulation experiments is well documented in the literature (3, 8, 9, 17). The simulation experiments provide the simulated data. The decision maker is concerned with three problems: The problem of stochastic convergence; the problem of size; and, the problem of motive (3). These problems are the basis of this component, which is shown in Figure 14.

The problem of motive concerns the accuracy with which the simulation experiment objectives are specified. Specifying the objectives as precisely as possible facilitates design selection. If the wrong design is chosen, the results will not be valid.

Stochastic convergence is the convergence of sample averages to population averages. The decision maker can solve this problem by increasing the sample size in three ways: (1) increase the simulation run length; (2) replicate the current run length; or (3) decrease the minimum time-unit. These methods are explained in detail in the literature (12, 10, 13). Monte Carlo techniques can also be used (3). The decision maker may not consider each of these methods in detail, but he should be aware of them during the experiment design.

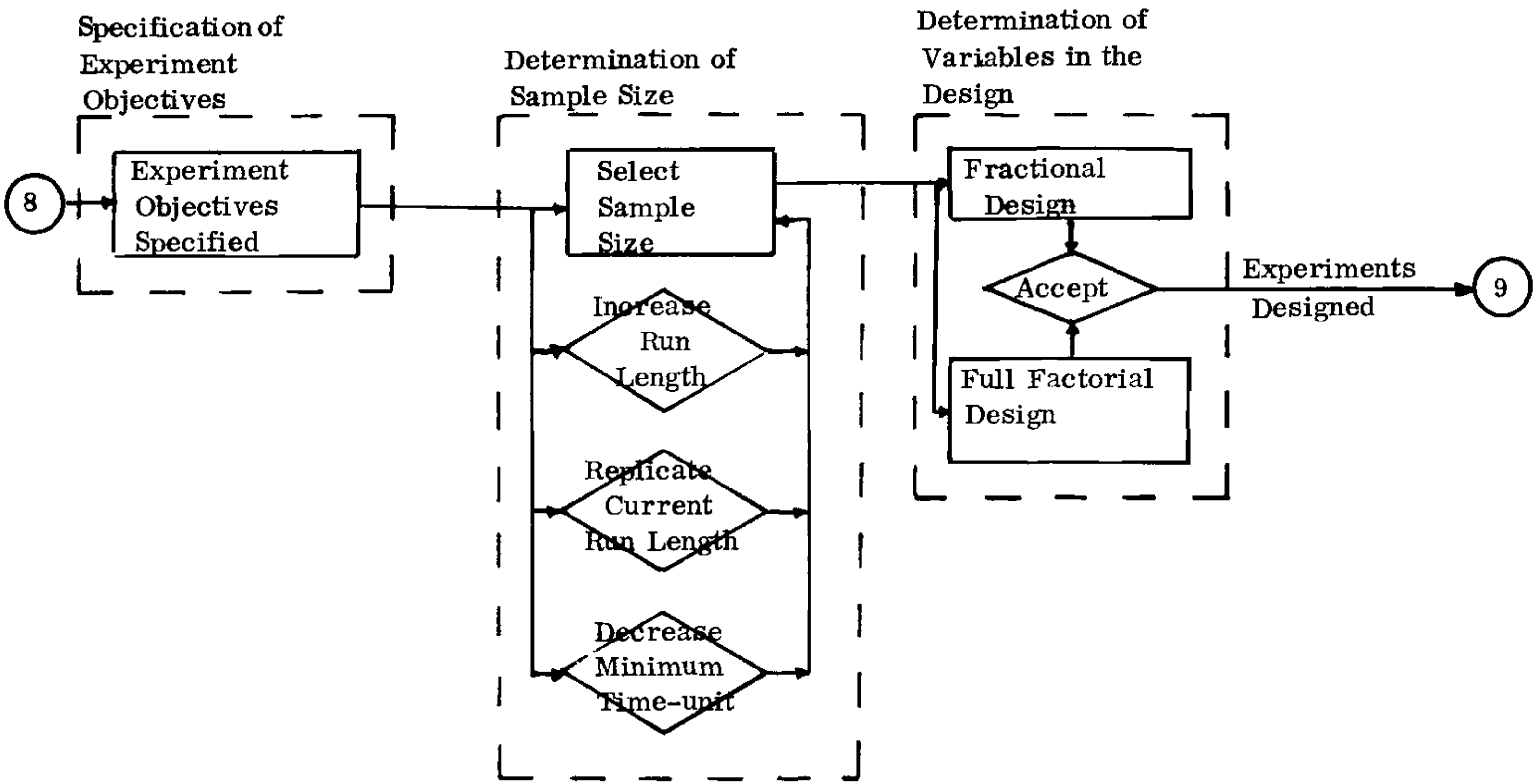

Figure 14. Design of Simulation Experiments--Components 8.

The problem of size concerns the number of factors in the experiment. When limiting the number of factors by constructing a fractional design, the trade-offs are the results achieved from the fractional design versus those achieved from a full factorial design. The costs of both alternatives include: (1) experiment planning time; (2) computer time required; and (3) data losses due to compounding effects.

Completion of the design makes the experiments ready for use or operation by the computer.

<u>Operation of the Computer Simulation Model--Component 9</u>

The operation of the computer simulation model is shown in Figure 15.

Operation of the computer simulation requires use of computer time. The decision maker must decide how to use the available time. Possible alternatives include: (1) to run only certain components; (2) to run the entire model; or (3) to replicate runs of one component. Each alternative's benefits are compared to the cost of operation of the alternative.

Scheduling allocated run time includes consideration of: (1) computer operating efficiency; and (2) the ability of the simulation to produce the desired results in the scheduled time. This is generally coordinated with data processing personnel. Scheduling computer time on a regular basis is necessary for repeated operation of the simulation.

<u>Analysis and Use of Simulation Results--Component 10</u>

The analysis and use of simulation results component is shown on

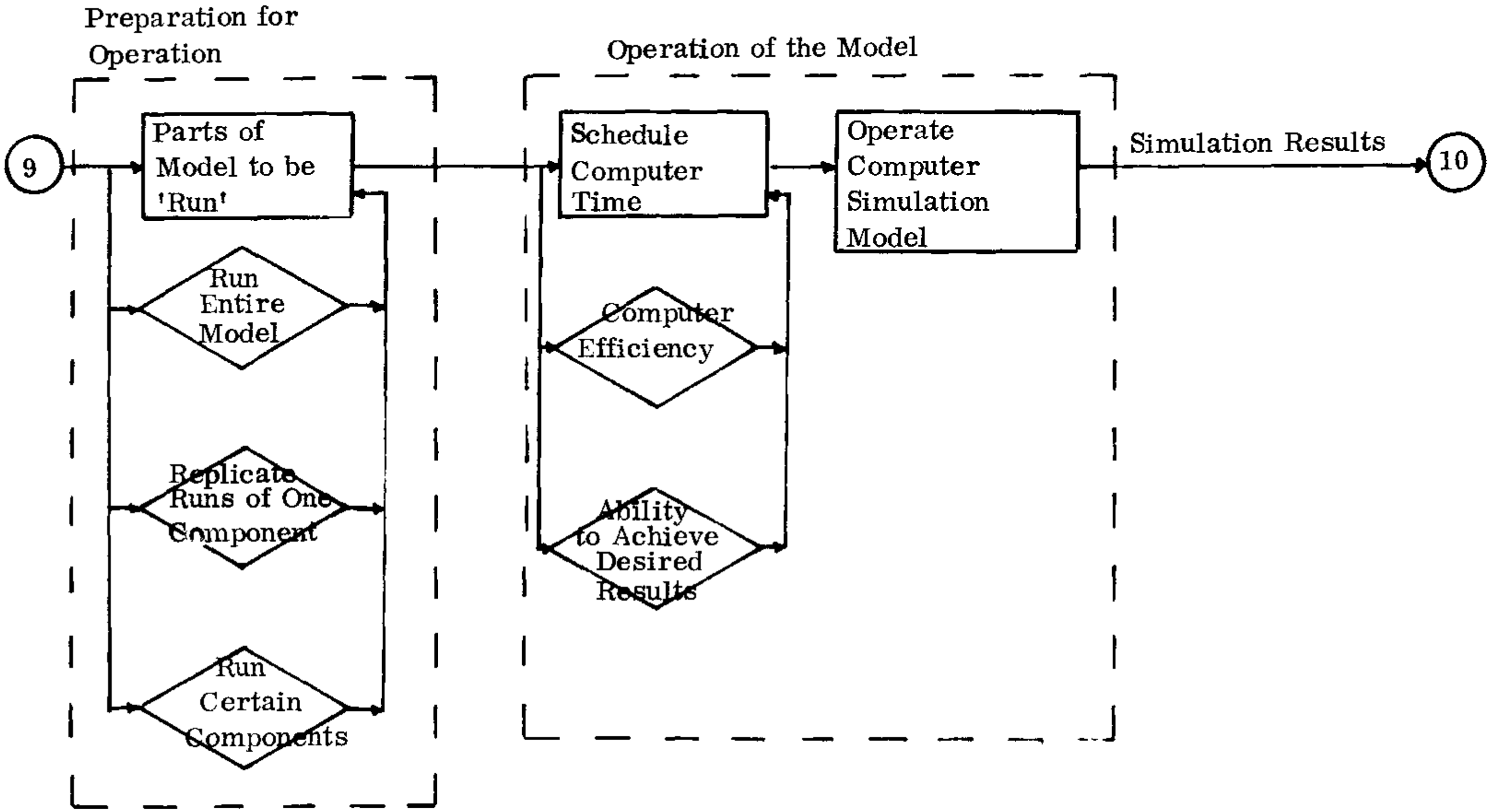

Figure 15. Operation of the Computer Simulation Model--Component 9.

Figure 16.

Operation of the computer simulation produces the data specified by Components 1 through 5, in the output formats encoded in Component 6. The simulation results, after collection and processing, are ready for testing. Selection and conduct of the tests is accomplished in the manner presented in Component 7. Rejection of the simulation results, after testing, may require that all or some of the preceding components be repeated. In general, Components 8 through 10 are repeated. The decision maker evaluates the reasons for rejection when deciding which components will be repeated. Acceptance of the test results means that the simulation data is ready for interpretation and use.

Interpretation of these results is the first step of implementation in the real world system. Because interpretation determines the real world system decision alternative to be supported, it should be done in an unbiased manner. If the decision maker forces the simulation results to support a pre-conceived decision, the value of the entire simulation can be lost.

After interpretation of the results, the decision maker implements the decision supported by the simulation data.

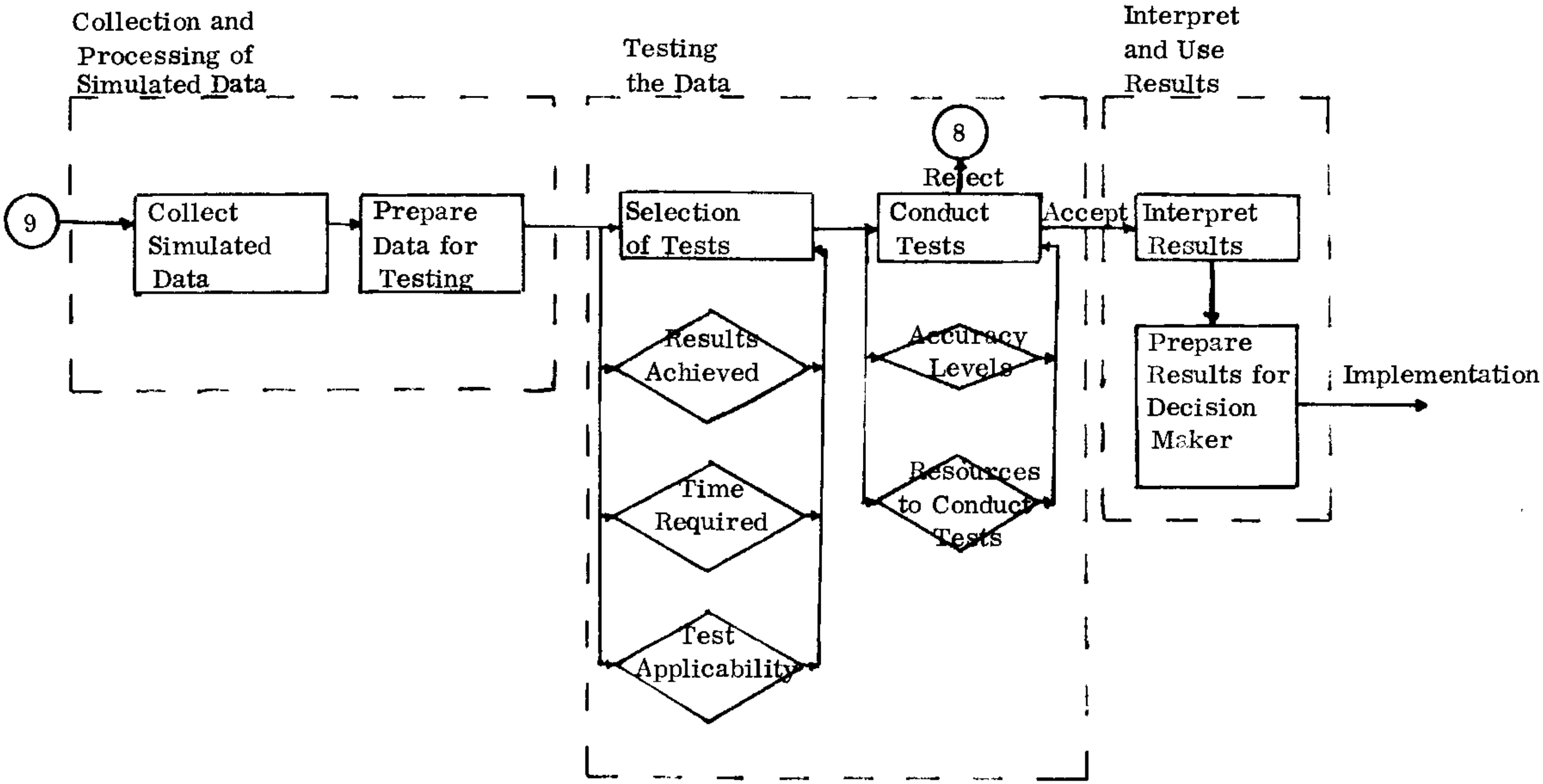

Figure 16. Analysis and Use of Simulation Results--Component 10.

CHAPTER IV

DECISION-FLOW DIAGRAM MODEL

The exhaustive model developed in Chapter III presented the components that organize and structure the computer simulation development process. The resultant simulation is shaped by the decisions made during the development process. The size and complexity of the simulations discussed in this research makes decision making about simulation alternatives similarly complex. This thesis presents a formal procedure for making simulation decisions, in the hope of providing a useful aid to supplement the manager's judgement and intuition.

The procedure selected to aid in the decision making is a version of the "decision-flow diagram or tree" (21). The decision-flow diagram helps the decision maker to reason and to act in a systematic manner in situations in which some events occur that cannot be predicted with certainty. This enables him to select the best strategy ("best" from an expected-value point of view) among the alternatives.

The decision-flow diagram presents, in chronological order, the alternatives available and the information known to the decision maker as he progresses through its various paths. The decision-flow diagram is a series of <u>decision forks</u> (decisions controlled by the decision-maker) and <u>chance forks</u> (decisions controlled by nature or chance). At each fork, distinct alternatives are available. Each alternative has an expected cost associated with it. The cost is incurred when that particular alternative is chosen. The alternatives connect the decision and chance forks to form paths through the decision-flow diagram. Whereas the decision maker selects the alternative at a decision fork, nature or chance, selects the alternative at a chance fork. Each alternative at a chance fork has a probability or chance of being selected by nature. Assignment of numerical values for these probabilities is difficult (21). Use of the probability values will be explained later.

In a decision-flow diagram, the decision forks are represented by squares, the chance forks are represented by circles, and the alternatives are represented by arrows that connect the forks.

Explanation of the Developed Decision-Flow Diagram

The decision-flow diagram provides structure for the decision making associated with the development of a computer simulation.

Each alternative presented at the decision and chance forks is the best of a group of possible choices for that alternative. For example, consider the Outside Special Purpose alternative at the Development Source and Language Selection Fork (Decision Fork 3), in which the decision-maker decides on the development source and the language to be used to translate the problem into a computer program (See Figure 17). The choices here are between a special-purpose and a general-purpose language and between in-house and outside-contract writing of the program. It is assumed that the best among the general-purpose languages, among the special-purpose languages, and among the possible outside contractors has already been selected before development of the decision-flow diagram. For this alternative, for example, the original choices may have been Company A using ALGOL, Company

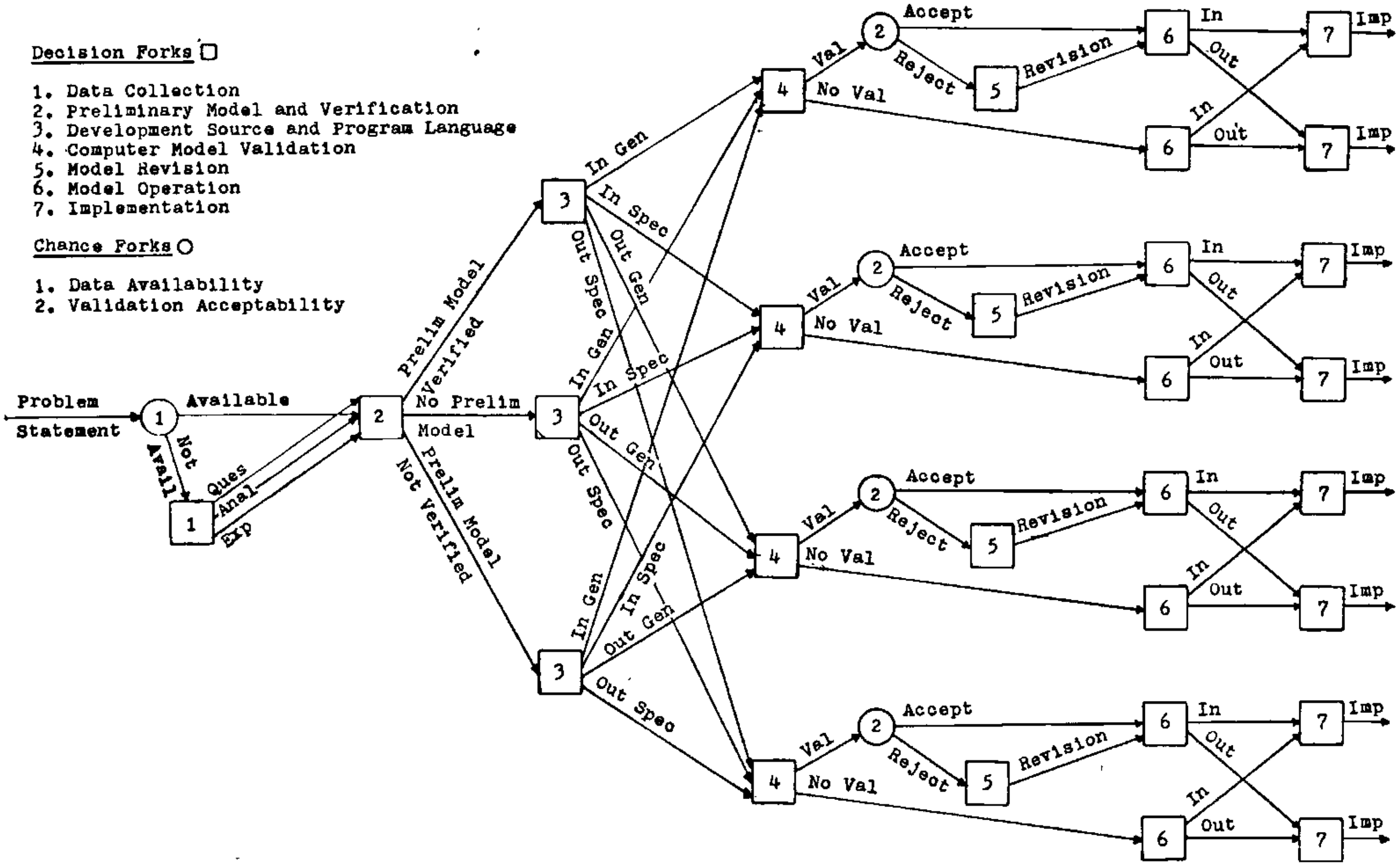

Figure 17. Decision-Flow Diagram for the Simulation Development Process.

B using FORTRAN, Company C using COBOL, and Company D using FORTRAN. From these choices, Company B using FORTRAN was selected as best and is the only version of this alternative given in the decision-flow diagram.

Explanation of the Forks in the Decision-Flow Diagram

The decision-flow diagram for the simulation development process is shown in Figure 17.

Data Availability Fork

Chance Fork 1 is the Data Availability Fork. At this fork, the simulation will have been initiated by the refined problem statement and chance controls the availability of the required data in the developing organization's data base. There are two alternative decisions available at this fork: the Available alternative, in which the data required for the problem's solution is in the developing organization's data base; and the Not Available alternative, in which the data required for the problem's solution is not available in the developing organization's data base.

Data Collection Fork

Decision Fork 1 is the Data Collection Fork. At this fork, the simulation will not have the required data available in the developing organization's data base, and the decision maker must decide on a method of data collection. There are three alternative decisions available at this fork: the Questionnaire alternative, in which a questionnaire is used to obtain the data required; the Analysis alternative, in which a form of engineering analysis is conducted to achieve the required data; and the Experiments alternative, in which experiments that provide the required data are performed.

Preliminary Model and Verification Fork

Decision Fork 2 is the Preliminary Model and Verification Fork. At this fork, the simulation will have the data collected and processed and the decision maker decides whether or not to use a preliminary model, and if a preliminary model is used, whether or not to verify its results. There are three alternative decisions available at this fork: the Preliminary Model Verified alternative, in which a preliminary model is used and the model's results are verified; the Preliminary Model Not Verified alternative, in which a preliminary model is used but the model's results are not verified; and the No Preliminary Model alternative, in which no preliminary model is used.

Development Source and Language Selection Fork

Decision Fork 3 is the Development Source and Language Selection Fork. At this fork, the decision maker decides on the development source and the language to be used to translate the problem into a computer program. There are four alternative decisions available at this fork: the In-house General Purpose alternative, in which the computer model is formulated by organization personnel using a general-purpose language; the In-house Special Purpose alternative, in which the computer program is formulated by organization personnel using a special purpose simulation language; the Outside General Purpose alternative, in which the computer program is formulated by an outside-contract firm using a general-purpose language; the Outside Special Purpose alternative, in which the computer program is formulated by an outside-contract firm using a special purpose simulation language.

Computer Model Validation Fork

Decision Fork 4 is the Computer Model Validation Fork. At this fork, the simulation will have a developed computer model. The decision maker decides whether or not to formally validate the computer model. There are two alternative decisions available at this fork: the Validation alternative, in which the computer model is validated; and the No Validation alternative, in which the computer model is not validated.

Validation Acceptability Fork

Chance Fork 2 is the Validation Acceptability Fork. At this fork, the simulation will have a computer model that has been validated, and nature controls the acceptability of this validation. There are two alternative decisions available at this fork: the Acceptable alternative, in which the validation results are accepted by the decision maker; and the Not Acceptable alternative, in which the validation results are not accepted by the decision maker.

Model Revision Fork

Decision Fork 5 is the Model Revision Fork. At this fork, the simulation will have validation results which are not accepted by the decision maker and the decision maker must revise the computer model. Usually the only alternative available at this fork is the Revision alternative, in which the decision maker revises the computer model to make it produce acceptable results. The extent of revision is, of course, variable.

Operation Source Fork

Decision Fork 6 is the Operation Source Fork. At this fork, the simulation computer model must be operated or 'run' and the decision maker decides on the source to operate the computer model. There are two alternative decisions available at this fork: the In-house alternative, in which the computer model is operated on a computer owned by the developing organization; and the Outside alternative, in which the computer model is operated on an outside-contract firm computer.

Implementation Fork

Decision Fork 7 is the Implementation Fork. At this fork, the simulation will have been operated and the decision maker decides on the manner of implementation of the simulation results. Usually the only alternative decision available at this fork is the Implementation alternative, in which the simulation results are implemented in the real world system by the decision maker.

Use of the Decision-Flow Diagram

Before the developed decision-flow diagram can be used, the expected cost of each alternative at all forks must be obtained. These expected costs result from work, time, personnel, and money expenditure estimates made by the developing organization for activities associated with alternatives accomplished within the developing organization. Bids or contract offers provide the cost data for alternatives involving firms other than the developing organization. These costs are placed on the respective alternatives.

Next, the probabilities associated with the chance fork alternatives must be specified. Raiffa presents several techniques for determining these probabilities (21).

The research of Abt Associates, Inc. indicates that for a typical simulation study, the probability of finding the required data already in useable form is about .56, and the probability of finding model validation results acceptable without requiring model revision is about .50 (1). This thesis uses these two probability assignments for illustrative purposes at Chance Forks 1 and 2 respectively.

Determination of the optimal strategy is the final step in the decision-making process. The goal of the decision maker is to minimize the expected cost of the entire simulation. Minimizing the expected cost is equivalent to maximizing the expected net benefits because it is assumed that every simulation which is considered acceptable for the given problem will provide the maximum benefits possible from a simulation. Thus, the decision maker examines and compares the various acceptable simulations for the given problem and selects the simulation with the minimum expected cost.

Justified arguments can be made in the support of the belief that decision making is not based solely on cost minimization or profit maximization. However, this research uses cost minimization as the goal of the decision maker for the following reasons:

1. Cost minimization is easily understood and can be directly applied by all users of the developed decision-flow diagram.
2. The case examples used in this research would provide only cost estimates for the simulations being developed.

The size of the simulations discussed in this thesis makes it necessary to consider the importance of <u>time</u> in the decision-making process. The time between successive decisions may be substantial. At the decision and chance forks, it may be necessary to compare differences in immediate costs against differences in future costs. Consideration of time and its effect on the decision-making process is particularly important when the translation, validation, operation, and implementation of a simulation will extend over a period of years. The alternatives can be put on a comparable basis by the use of the Present Worth technique. This technique says:

Given a model that will be used for N years at an expected cost of $\bar{A}$ dollars per year, the present worth of the cost is: (25)

$$P = \frac{\bar{A}}{r}\left(1 - e^{-rN}\right).$$

where $r = \ln(1 + i)$

i = annual compound interest rate

The interest rate, i, varies and is established by the individual organization.

The optimal decision strategy is achieved by working backwards from the tips of the diagram. Two separate computational devices are used: (21)

1. An averaging-out process at each chance fork.
2. A choice process selecting the minimum cost path at each decision fork.

The averaging-out process uses the probability assigned to each chance fork alternative and the cost of the respective alternative path. The probability that chance selects the alternative is multiplied by the path cost. This is repeated for all alternatives at the chance fork. The values obtained for all of the alternatives are summed and this value is placed at the chance fork. Consider the example chance fork presented in Figure 18.

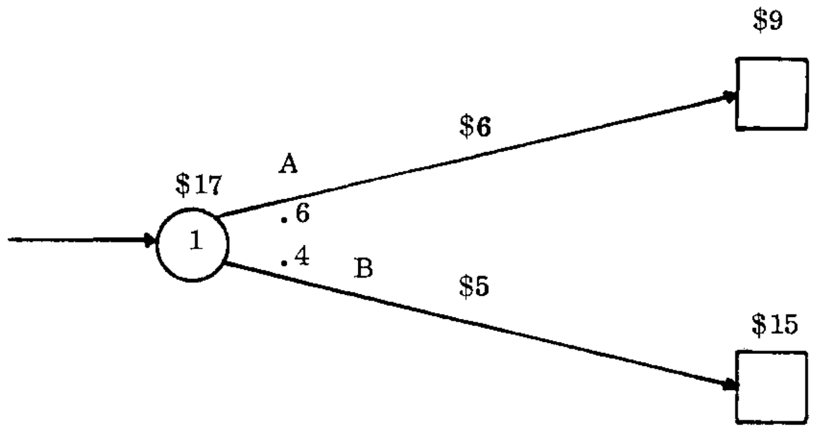

Figure 18. Example Chance Fork in a Decision-Flow Diagram.

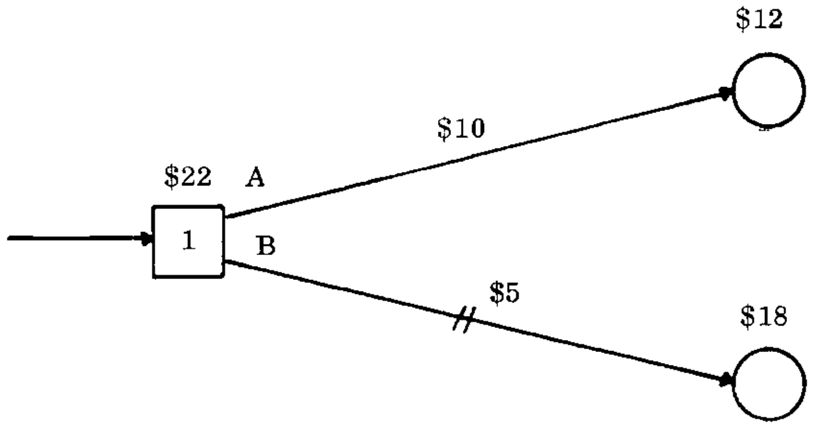

Figure 19. Example Decision Fork in a Decision-Flow Diagram.

The path cost is the sum of the alternative cost and the cost at the fork that the alternative precedes. The cost of Path A is:

$$\$6 + \$9 = \$15$$

This cost is multiplied by the probability that Alternative A is selected by chance, which is .6. This yields:

$$.6 \times \$15 = \$9$$

This process is repeated for Path B. This produces:

$$\$5 + \$15 = \$20$$

$$.4 \times \$20 = \$8$$

The path costs are summed:

$$\$9 + \$8 = \$17$$

This value, \$17 , is placed at Chance Fork 1.

At each decision fork, one alternative must be selected. When the decision maker is at the decision fork, he looks down the alternative paths and selects the path of lowest cost. The path cost is the sum of the alternative cost and the cost at the fork that the alternative leads to. Consider the example decision tree presented in Figure 19. At Decision Fork 1, Path A or Path B must be chosen by the decision maker. The cost of Path A is:

$$\$10 + \$12 = \$22$$

The cost of Path B is:

$$\$5 \quad + \ \$18 \quad = \ \$23$$

Path A is the least expensive, and is the path selected. The value of \$22 is placed at Decision Fork 1. The decision maker who selects Alternative B because it is less expensive than Alternative A makes an error. He fails to look ahead past the alternative to the cost of the remainder of the path.

These two basic tools are used during the fold back procedure which determines the optimal strategy selected by the decision maker. Starting at the tips, or ends, of the decision-flow diagram, the decision maker works backwards, or from right to left, through the diagram. The path cost is placed at the appropriate decision or chance fork using the computational techniques previously described. The alternative paths not selected are marked with double vertical slashes as shown in Figure 19 (Path B).

Utilizing the developed decision-flow diagram and the computational techniques provided, the simulation developer can structure his decision-making process and determine his optimal decision strategy based on the costs and probabilities applicable to his particular simulation.

CHAPTER V

APPLICATIONS OF THE DECISION-FLOW DIAGRAM

This chapter presents the application of the decision-flow diagram developed in Chapter IV to case examples. The two organizations selected were developing computer simulations of the size and type discussed in this research. Information about the development of each simulation was obtained from the individual responsible for the simulation's development. Each individual's responses to a prepared list of questions were recorded on tape. From these responses, cost data required for use on the decision-flow diagram was obtained.

Both organizations were promised anonymity by this author and will be referred to as "Organization B" and "Organization R" respectively.

Case Example 1

Organization B was developing a simulation of an information flow network. The organization had no operating procedures or policies, such as not permitting the use of a contract firm for research or planning projects, which would limit the decision-making flexibility during the development of the simulation. Thus, the entire decision-flow diagram, as developed in Chapter IV, is applicable to Organization B. This is shown in Figure 20.

Reducing the number of alternatives considered at Decision Fork 1 and Decision Fork 3 did, however, limit the size of the decision-flow diagram. At

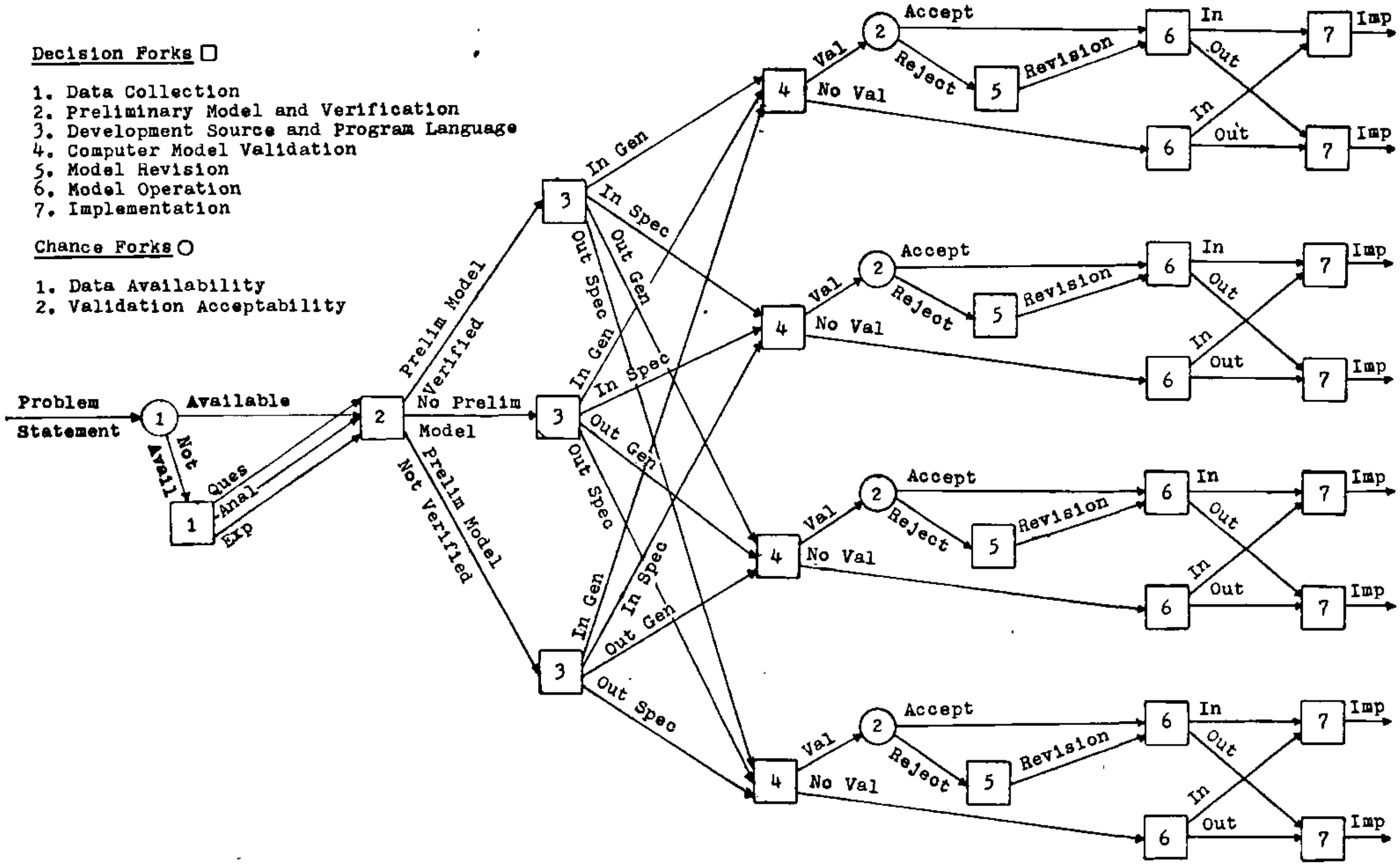

Figure 20. Unconstrained Decision-Flow Diagram of Organization B.

Decision Fork 1, only experiments were considered as an acceptable method of data collection. The best, from an economic comparison, was chosen. At Decision Fork 3, the special purpose simulation language chosen to be the alternative, In-house Special Purpose, with this alternative ultimately being selected, could not be used, or run, on the organization's computer. Program development by a contracted firm, using a general purpose language or a special purpose simulation language, was not considered. This limitation of the alternatives considered at these decision forks reduced the size of the decision-flow diagram to that shown in Figure 21.

This decision-flow diagram indicates only two decision forks with more than one alternative to choose from; Decision Fork 2, Preliminary Model and Verification, and Decision Fork 4, Computer Model Validation. Because a previously formulated small manual network configuration provided the information generally obtained from a preliminary model, the decision maker decided against the use of a preliminary model, as such. Thus, the alternative, No Preliminary Model, was selected at Decision Fork 2.

At Decision Fork 4, Computer Model Validation, the decision maker chose to validate the computer model. Sophisticated statistical validation techniques were not employed, but the model results were compared to the real world data. From this comparison, the model was accepted as valid and operated without revision.

The path formulated by Organization B through the decision-flow diagram, and the alternative costs, is shown in Figure 22.

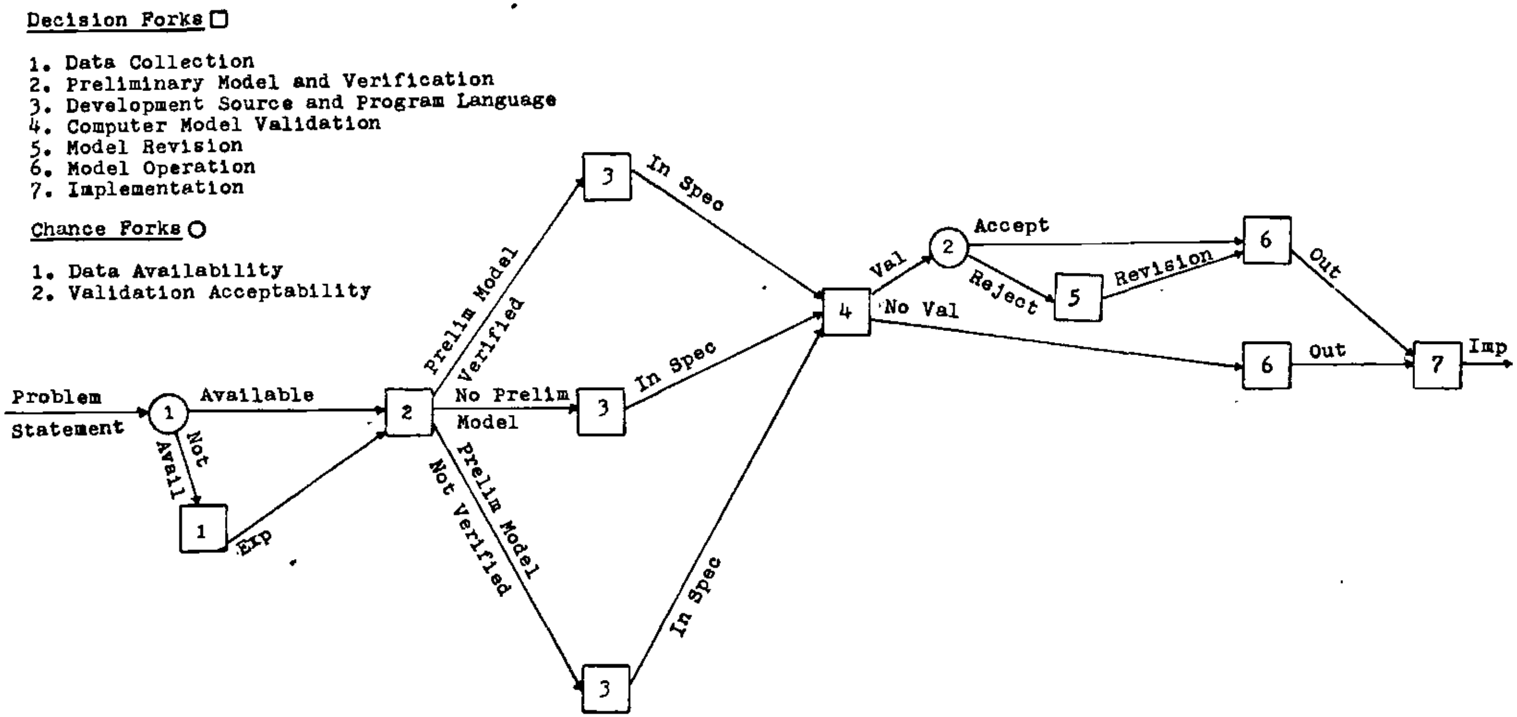

Figure 21. Reduced Decision-Flow Diagram of Organization B.

Decision Forks

1. Data Collection
2. Preliminary Model and Verification
3. Development Source and Program Language
4. Computer Model Validation
5. Model Revision
6. Model Operation
7. Implementation

Chance Forks

1. Data Availability
2. Validation Acceptability

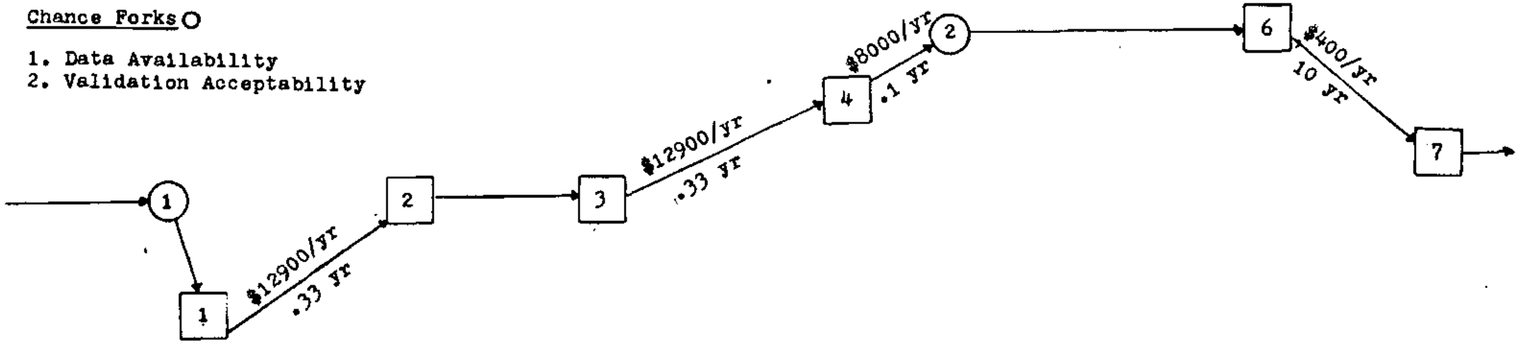

Figure 22. Organization B Path Through Decision-Flow Diagram.

Case Example 2

Organization R was developing a simulation of a transportation system network. The organization's operating procedures or policies did not permit computer program development or program operation by a contracted firm. This reduced the size of the decision-flow diagram developed in Chapter IV to that shown in Figure 23.

This decision-flow diagram was not further reduced in size by limitation of the number of alternatives considered at any decision fork.

Because the required data was available in the organization's data base, Decision Fork 2, Preliminary Model and Verification, was the first decision made by the decision maker. A previously developed computer simulation model provided the information generally gained from a preliminary model. Thus, the decision maker chose the alternative, No Preliminary Model, at Decision Fork 2.

Both alternatives at Decision Fork 3, Development Source and Language Selection, were considered. Use of a special purpose simulation language by organization personnel, In-house Special Purpose, was an apriori selection. However, consideration was given to the alternative, In-house General Purpose. However, flow-charting and encoding were the characteristics primarily considered. Operation of the developed program in the selected language was not considered in this decision. The decision-flow diagram, and the fold back procedure, allows the decision maker to consider operation costs when the development source and language is selected.

The decision maker decided against validation of the computer model at

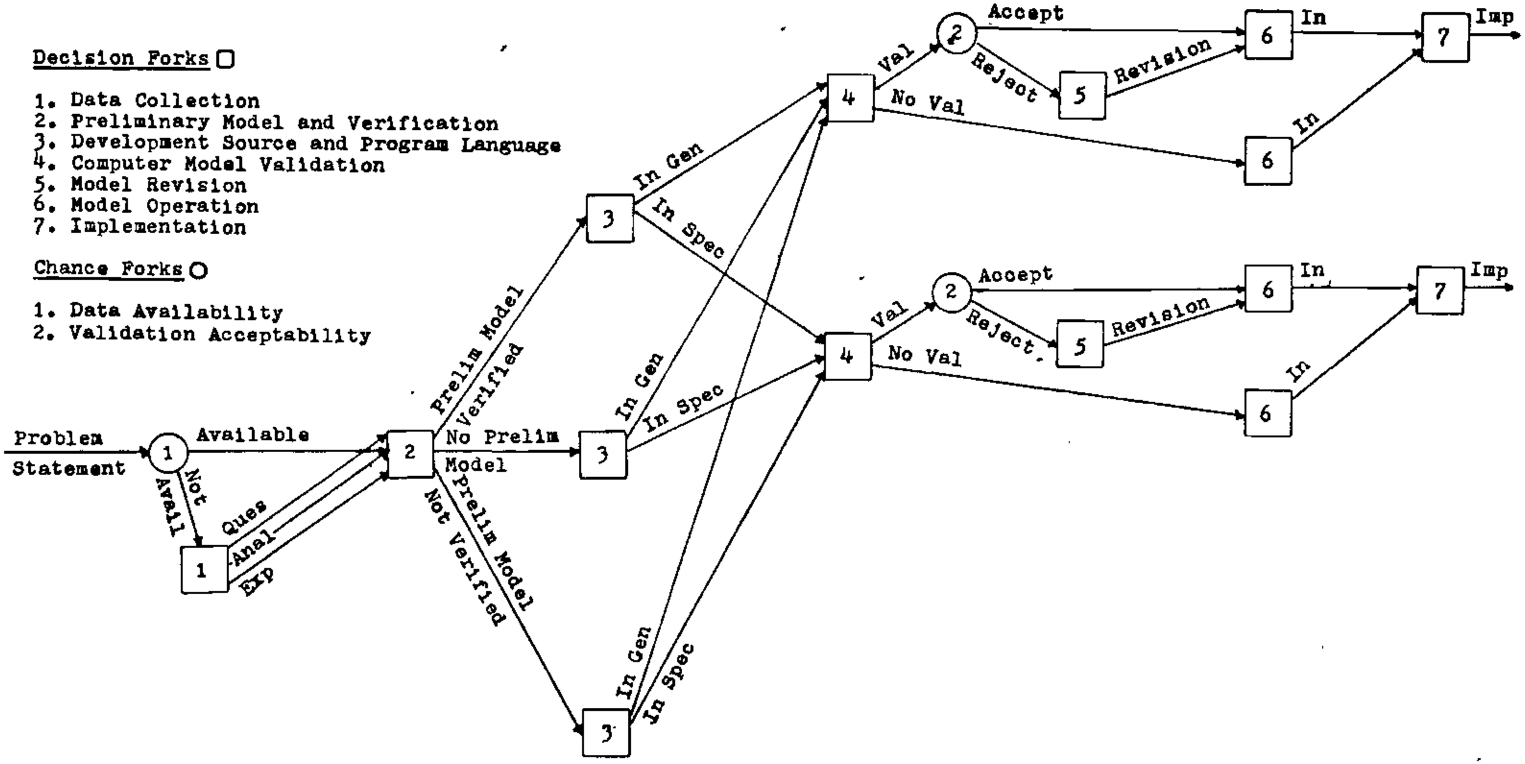

Figure 23. Constrained Decision-Flow Diagram of Organization R.

Decision Fork 4. This decision designated the remainder of the path through the decision-flow diagram.

The path formulated by Organization B through the decision-flow diagram is shown on Figure 24.

Analysis of Case Examples

Both organizations (Organization B and Organization R) have established a clearly defined path through the decision-flow diagram. No attempt will be made by this author to justify these paths as optimal nor to degrade them as examples of poor decision making. The paths of both organizations, and the associated costs, will be used to demonstrate the use of the Present Worth technique in comparing the decision alternatives in the decision-flow diagram.

As explained in Chapter IV, the impact of time should be considered in economic decision making. The decision at Decision Fork 3, and the comparison of the alternatives at this fork, will be used to demonstrate the use of the Present Worth technique. For ease of explanation, the number of decision alternatives at Decision Fork 3 are reduced to the following:

1. Special Purpose Simulation Language.
2. General Purpose Language.

The selection of a computer language and the methods used to compare languages is not an exact process because of the variety and range of possible evaluation criteria (18). The criteria used in this explanation is cost minimization, the previously stated goal of the decision maker responsible for the simulation development.

Decision Forks □

1. Data Collection
2. Preliminary Model and Verification
3. Development Source and Program Language
4. Computer Model Validation
5. Model Revision
6. Model Operation
7. Implementation

Chance Forks ○

1. Data Availability
2. Validation Acceptability

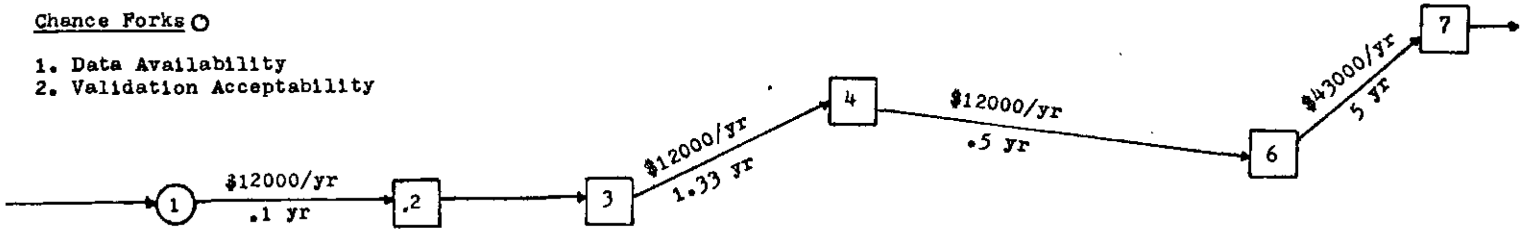

Figure 24. Organization R Path Through Decision-Flow Diagram.

The costs associated with these alternatives include: Translation cost; validation cost; model operation cost; and implementation cost. Because implementation of the simulation results does not depend on the type of language used, but is basically a function of the managerial abilities of the analyst, the time and cost of implementation are considered to be equal for both decision alternatives (6). The cost of the alternatives are put on a comparable basis by use of the Present Worth technique. The decision maker compares the alternative costs and selects the minimum cost alternative. This decision is shown (in decision-flow diagram format) in Figure 25.

Generally, the time required for problem translation, and the cost, is greater when a general purpose language is used than it is when a special purpose simulation language is used. The computer model operation time, and the cost, is generally greater when a special purpose simulation language is used than it is when a general purpose language is used.

A more definite comparison of the two language types is required for use in this thesis. To obtain the comparative relationships for translation time, validation time, and model operation time, an informal survey of several computer firms was conducted. This survey produced the following approximate relationships, which are used in this thesis:

1. The translation time, and cost, is approximately two times greater for a general purpose language than for a special purpose simulation language.
2. The model operation time, and cost, is approximately eight times

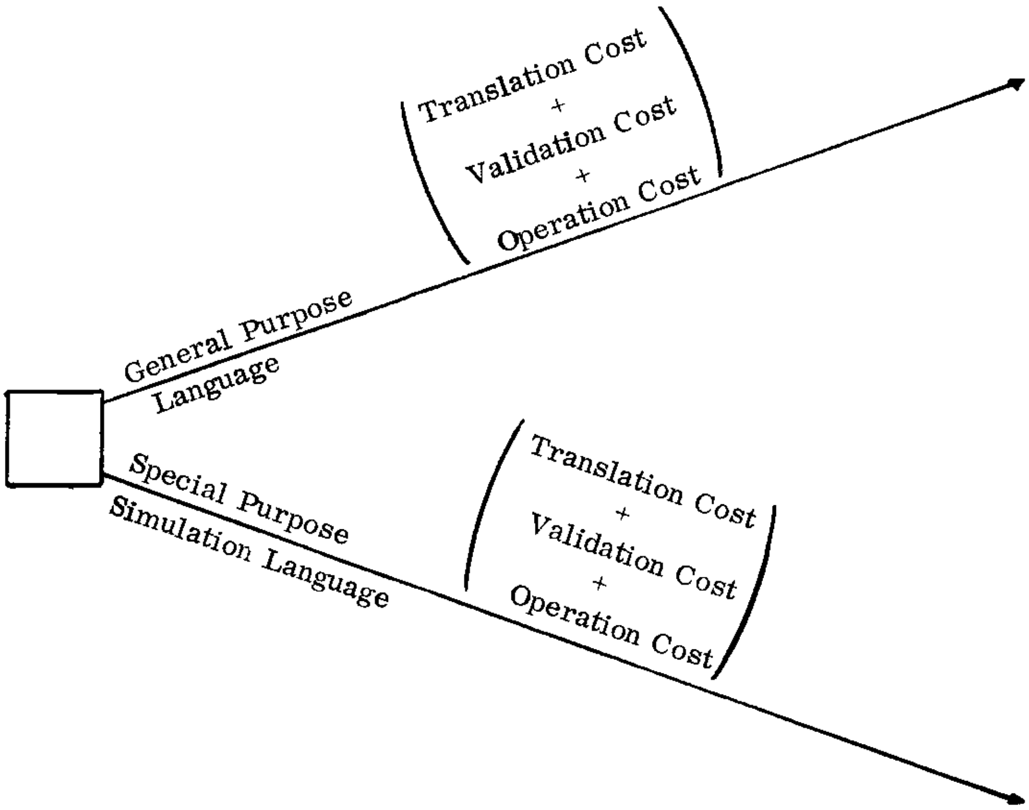

Figure 25. Decision Fork Used in Case Example Analysis.

greater for a special purpose simulation language than for a general purpose language.

3. The validation time, and cost, is approximately 1.5 times greater for a special purpose simulation language than for a general purpose language.

The annual compound interest rate is assumed to be ten per cent (10%) for both organizations.

Organization B

The simulation model developed by Organization B used a special purpose simulation language. The path of Organization B through the decision-flow diagram (as was shown in Figure 22) displayed the time and cost for translation, validation, and model operation. The times and costs, to be used in determining the cost of this alternative are:

1. Translation cost and time
    a. Cost $12,900/year
    b. Time .33 year
2. Validation cost and time
    a. Cost $8000/year
    b. Time .1 year
3. Model operation cost and time
    a. Cost $400/year
    b. Time 10 years

The total cost of this alternative, using the Present Worth technique is now computed. The present worth of the total cost is:

$$P(\text{Cost}) = P(\text{Translation Cost}) + P(\text{Validation Cost} + P(\text{Model Operation Cost})$$

$$P(\text{Cost}) = \frac{12900}{r}\left[1 - e^{-r(.33)}\right] + \frac{8000}{r}\left[e^{-r(.33)} - e^{-r(.43)}\right]$$

$$+ \frac{400}{r}\left[e^{-r(.43)} - e^{-r(10.43)}\right]$$

where $i = .1$

$$r = \ln(1 + i) = \ln(1 + .1) = .09531$$

$$P(\text{Cost}) = \frac{12900}{.09531}\left[1 - e^{-.03145}\right] + \frac{8000}{.09531}\left[e^{-.03145} - e^{-.04098}\right]$$

$$+ \frac{400}{.09531}\left[e^{-.04098} - e^{-.99408}\right]$$

$$P(\text{Cost}) = 135347.81\,(1. - .9690) + 83936.63(.9690 - .9598) + 4196.83(.9598 - .3701)$$

$$P(\text{Cost}) = 4196 + 772 + 2475 = \$7443.$$

To determine the present worth of the cost of the general purpose language alternative, the translation cost, the validation cost, and the model operation cost are adjusted by the previously presented approximate relationships.

The costs and times to be used in determining the cost of this alternative are:

1. Translation cost and time.

    a. Cost \$25,800/year

    b. Time .33 year

2. Validation cost and time.

    a. Cost \$12,000/year.

    b. Time .1 year.

3. Model Operation cost and time.

    a. Cost \$50/year.

    b. Time 10 years.

P(Cost) = P(Translation Cost) = P(Validation Cost +

P(Model Operation Cost)

$$P(\text{Cost}) = \frac{25800}{r}\left[1 - e^{-r(.33)}\right] + \frac{12000}{.09531}\left[e^{-r(.33)} - e^{-r(.43)}\right]$$

$$+ \frac{50}{r}\left[e^{-r(.43)} - e^{-r(10.43)}\right]$$

where $i = .1$

$r = \ln(1 + i) = \ln(1 + .1) = .09531$

$$P(\text{Cost}) = \frac{25800}{.09531}\left[1 - e^{-.03145}\right] + \frac{12000}{.09531}\left[e^{-.03145} - e^{-.04098}\right]$$

$$+ \frac{50}{.09531}\left[e^{-.04098} - e^{-.99408}\right]$$

$$P(\text{Cost}) = 270695.62\ (1 - .9690) + 125904.94(.9690 - .9598)$$

$$+ 524.60(.9598 - .3701)$$

P(Cost) = 8392. + 1158 + 309. = $9859.

Based on the cost minimization goal, the decision maker would select the special purpose simulation language alternative.

Organization R

The simulation model developed by Organization R used a special purpose simulation language. The path of Organization R through the decision-flow diagram (as shown in Figure 24) displayed the time and cost for translation, validation, and model operation. The times and costs to be used in determining the cost of this alternative are:

1. Translation cost and time
   a. Cost $12,000/year
   b. Time 1.33 years
2. Validation cost and time
   a. Cost $12,000/year
   b. Time .5 years
3. Model Operation cost and time
   a. Cost $43,000/year
   b. Time 5 years

P(Total Cost) = P(Translation Cost) + P(Validation Cost) + P(Model Operation Cost)

$$P(\text{Cost}) = \frac{12000}{r}\left[1 - e^{-r(1.33)}\right] + \frac{12000}{r}\left[e^{-r(1.33)} - e^{-r(1.83)}\right]$$

$$+ \frac{43000}{r}\left[e^{-r(1.83)} - e^{-r(6.83)}\right]$$

$$P(\text{Cost}) = \frac{12000}{.09531}\left[1 - e^{-.12676}\right] + \frac{12000}{.09531}\left[e^{-.12676} - e^{-.17442}\right]$$

$$+ \frac{43000}{.09531}\left[e^{-.17442} - e^{-.65097}\right]$$

$$P(\text{Cost}) = .125904.94(1 - .8809) + .125904.94(.8809 - .8399)$$

$$+ 451159.37(.8399 - .5215)$$

$$P(\text{Cost}) = 14995 + 5162 + 143649 = \$163,770.$$

To determine the present worth of the cost of the general purpose language alternative, the translation time, the validation time, and the model operation time are adjusted by the previously presented approximate relationships.

The costs and times to be used in determining the cost of this alternative are:

1. Translation cost and time
   a. Cost $24,000/year
   b. Time 1.33 years
2. Validation cost and time
   a. Cost $18,000/year
   b. Time .5 years

3. Model Operation cost and time

a. Cost \$5375/year

b. Time 5 years

$$P(\text{Cost}) = \frac{24000}{r}\left[1 - e^{-r(1.33)}\right] + \frac{18000}{r}\left[e^{-r(1.33)} - e^{-r(1.83)}\right]$$

$$+ \frac{5375}{r}\left[e^{-r(1.83)} - e^{-r(6.83)}\right]$$

$$P(\text{Cost}) = \frac{24000}{.09531}\left[1 - e^{-.12676}\right] + \frac{18000}{.09531}\left[e^{-.12676} - e^{-.17442}\right]$$

$$+ \frac{.5375}{.09531}\left[e^{-.17442} - e^{-.65097}\right]$$

$$P(\text{Cost}) = 251809.88(1. - .8809) + 188857.41(.8809 - .8399)$$

$$+ 56394.92(.8399 - .5215)$$

$$P(\text{Cost}) = 29991 + 7743 + 17956 = \$55,690.$$

Based on the cost minimization goal, the decision maker would select the general purpose language alternative.

The above explanation presented the use of the present worth technique to compare alternatives in the decision-flow diagram. The results, because of the use of the approximate relationships between the costs of a general purpose language and those of a special purpose simulation language, can not be accepted as valid. However, the reader should be able to recognize that the decision-flow diagram does structure the decision-making process, while allowing the decision maker accurately to compare the decision alternatives.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

The conclusions of this research are:

1. The additional cost of applying the decision-flow method developed herein seems to be not more than about two man-months (perhaps $4000 to $5000) over the cost of making decisions in the traditional way. As seen in Chapter V, the probable savings in using a structural decision-making process are an order of magnitude higher than this. Hence, it seems well worth while to use the decision-flow method in making simulation development decisions for large-scale simulations.
2. Data gathered from the two organizations, which was used to validate the decision-flow diagram, verified the data collection method as applicable to large simulations. The narrative history method used to gather the relevant information for the decision-flow diagram appears readily useable by organizations developing simulation studies, regardless of whether the formal diagram itself is used.
3. There exist organizations developing simulations which have tended to ignore the economics associated with certain stages of the simulation development, causing them to incur unnecessary costs.

It is recommended that further research be done to exploit and expand the work of this thesis, specifically:

1. Develop data-based methods of estimating the probabilities of the chance outcomes in the decision-flow diagram (the probability data will be available in the needed form and the probability that the simulation mechanism will need substantial revision due to weaknesses diagnosed in the validation procedure) and of estimating all the costs.
2. Test the assumption that every simulation method which is considered acceptable for a given problem will provide the maximum implementation benefits possible from any simulation method. If necessary, revise the decision-flow method to allow varying benefits.
3. Repeat the validation of the decision-flow method by using it on further actual applications.
4. Extend the decision-flow method to smaller-scale simulations, where computer time and programming time are minor considerations and where such low-power languages as HOCUS and BASIC might be appropriate. In this realm the time discounted expected-cost approach may become inappropriate or unwieldly, since some small-scale simulations are justifiable mainly for the added insights given to a manager as to how his system reacts to changes.

BIBLIOGRAPHY

1. Abt Associates, "Survey of the State of the Art: Social Political, and Economic Models and Simulation," Technology and the American Economy, Vol. 5, Cambridge, Mass., 1966.

2. Appelbaum, Melvin, "Seven Steps to Simulation," Modern Data, Vol. 3, July, 1970.

3. Burdick, D. S. and Thomas Naylor, "Design of Computer Simulation Experiments for Industrial Systems," Communications of the Association for Computing Machinery, Vol. 9, No. 5, May, 1966.

4. Chubb, Bruce A., "Economic Evaluation of the CSMP Digital Simulation Language," Simulation, Vol. 14, March, 1970.

5. Cox, J. G. and J. H. Mize, Essentials of Simulation, Prentice-Hall, Englewood Cliffs, N.J., 1968.

6. Emshoff, James R. and Roger L. Sission, Design and Use of Computer Simulation Models, Macmillian Company, 1970.

7. Evans, George, Georgia Sutherland and Graham Wallace, Simulation Using Digital Computers, Prentice-Hall, Englewood Cliffs, N.J., 1967.

8. Fishman, George, "Digital Computer Simulation: Estimating Sample Size," Rand Corp., August, 1969.

9. Fishman, George, "Digital Computer Simulation: The Allocation of Computer Time in Comparing Simulation Experiments," Rand Corp., 1967a.

10. Fishman, George and Philip J. Kiviat, "Spectral Analysis of Time Series Generated by Simulation Models," Rand Corp., February, 1968.

11. Fried, Louis, "How to Analyze Computer Project Costs," Computer Decisions, Vol. 3, August, 1971.

12. Gafarian, A. V. and C. I. Ancker, "Mean Value Estimation from Digital Computer Simulation," System Development Corp., Santa Monica, Calif., 1965.

13. Geisler, Murray A., "The Sizes of Simulation Samples Required to Compute Certain Inventory Characteristics with Stated Precision and Confidence," Management Science, Vol. 10, January, 1964.

14. Gnugnoli, Giuliano and Herbert Maisel, Simulation of Discrete Stochastic Systems, Science Research Associates, Inc., Chicago, Ill., 1972.

15. Gordon, Geoffrey, Systems Simulation, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1969.

16. Hillier, Frederick S. and Gerald J. Leiberman, Introduction to Operations Research, Holden-Day, Inc., San Francisco, Calif., 1967.

17. Kiviat, Philip J., "Digital Computer Simulation: Modeling Concepts," Rand Corp., Santa Monica, Calif., August, 1967.

18. Lubian, John F. and Daniel Teichroew, "Computer Simulation: Discussion of the Technique and Comparison of Language," Communications of the Association for Computing Machinery, Vol. 9, No. 10, October, 1966.

19. Maynard, H. B., Industrial Engineering Handbook (Third Edition), McGraw-Hill Book Company, New York, 1971.

20. Naylor, Thomas H., et. al., Computer Simulation Techniques, Thomas H. Naylor, John Wiley & Sons, New York, 1966.

21. Raiffa, Howard, Decision Analysis: Introductory Lectures on Choices Under Uncertainty, Addison-Wesley, Reading, Mass., 1970.

22. Reitman, Julian, Computer Simulation Applications, Wiley-Interscience, New York, 1971.

23. Schmidt, J., and R. E. Taylor, Simulation and Analysis of Industrial Systems, Richard C. Irwin, Inc., Homewood, Ill. 1970.

24. Tocher, K. D., The Art of Simulation, Van Nostrand, Princeton, N.J., 1963.

25. Buck, James R. and Thomas W. Hill, Jr., "Laplace Transforms for the Economic Analysis of Deterministic Problems in Engineering," The Engineering Economist 16, No. 4, 1971, p. 247-263.